ULAM'S CONJECTURE IS TRUE FOR CONNECTED GRAPHS*

Raju Renjit. G,

rajurenjitgrover@yahoo.com, Grover house, Tripunithura, Eranakulam, Kerala, India.

Abstract

In this endeavor, we prove that Ulam's conjecture is true for connected graphs. Initially, we present a new algorithm for graph isomorphism. We then transform the statement of the conjecture to the newly developed algorithm.

---

[*] Thank You JESUS.

Contents

1 Introduction 3

2 Isomorphism algorithm 13
  2.1 Basic automata theory . . . 13
  2.2 Examples . . . 51
    2.2.1 First example . . . 53
    2.2.2 Second example . . . 58
  2.3 Assumptions, notations and definitions . . . 76
  2.4 Proving the algorithm . . . 185

3 Ulam's conjecture 236
  3.1 Sketch of proof . . . 236
  3.2 Definitions . . . 239
  3.3 Proving the conjecture . . . 272

1 Introduction

In

- This
    - ○ Endeavor,
        - ▷ We
            - – Show that:

“Ulam’s-conjecture”

- Is:

“true”

for

- Finite
    - ○ Connected
    - ○ Unlabeled
    - ○ And undirected:
        - ▷ Graphs

with

- No:
    - ○ Multi edges
        - ▷ And
            - – Self loops.
- If:

$$G$$

- Is:

“a graph,”

- Then:

$$G \quad - \quad v_i$$

is

- Defined
  - To:
    - ▷ Be

the

- Graph
  - Obtained
    - ▷ By:
      - Deleting

the

- Vertex:

$$v_i.$$

- The
  - Conjecture:
    - ▷ Can

be

- Stated
  - As,
    - ▷ Let:

$$G \qquad \textit{and} \qquad H$$

be

- Graphs,
    - And
        - Let:

$$G \quad \textit{have}: \quad p \quad \textit{vertices}: \quad v_i,$$

$$H \quad \textit{have}: \quad p \quad \textit{vertices}: \quad u_i,$$

- Where:

$$p \geqslant 3.$$

- Then
    - If
        - For
            - Each:

$$i,$$

the

- Subgraphs:

$$G_i = G - v_i$$

$$H_i = H - u_i$$

are

- Isomorphic,
    - Then:

$$G \quad \textit{and} \quad H$$

are

- Isomorphic.
    - See

    - Harary [2]:
      - For details.

- Initially,
  - For
    - The:
      - Sake

of

- Proving
  - The:
    - Conjecture,

we

- Give
  - A new:
    - Algorithm

for

- Graph
  - Isomorphism.
    - We:
      - Then transform

the

- Statement
  - Of
    - The:
      - Conjecture

to

- That
    - Algorithm.
        - But:
            - Even though,

we

- Prove
    - The:
        - Conjecture

for

- Undirected
    - And
        - Unlabeled:
            - Graphs,

we

- Do
    - It
        - By:
            - Generating

a

- Set
    - Of
        - Automata:
            - From

the

- Original

    - ○ Graph.
        - ▹ And:
            - – So

in

- Sub section 2.1,
    - ○ We give
        - ▹ A brief:
            - – Introduction

to

- Automata
    - ○ Theory.
        - ▹ Let:

$$G$$

- Be:

“a graph.”

- Then:

$$V(G) \qquad \textit{and} \qquad E(G)$$

are

- The
    - ○ Set
        - ▹ Of:

vertices *and* *edges*

- In:

$$G.$$

- Alternatively:

$$G$$

can

- Be
    - Written
        - As:

$$(V(G), \quad E(G)).$$

- Also
    - If:

$$G_a \qquad \textit{and} \qquad G_b$$

are

- Graphs,
    - Then:

$$G_a \quad \subset \quad G_b$$

- Means:

$$G_a$$

is

- A subgraph
    - Of:

$$G_b.$$

- Also:

$$G_a \quad \cong \quad G_b$$

- Means:

$$G_a \qquad \textit{and} \qquad G_b$$

are

- Isomorphic.
    - An
        - Undirected:
            - Edge

between

- Two
    - Vertices,
        - Say:

$$v_i \qquad \textit{and} \qquad v_j$$

is

- Written
    - As:

$$(v_i, \quad v_j).$$

- And
    - A directed
        - Edge
            - From:

$$v_i \qquad \textit{to}: \qquad v_j$$

- Labeled:

$$\sigma$$

is

- Written
  - As:

$$\langle v_i, \quad v_j, \quad \sigma\rangle.$$

- Also:

$$\mathbf{N}$$

is

- Defined
  - To:
    - Be

the

- Set of
  - All
    - Positive-integers:
      - Including zero.
- Also:

$$\mathbf{P}(S)$$

is

- Defined
  - To:
    - Be

the

- Power

  - Set
    - Of:

$$S.$$

- Also:

$$\in, \qquad \subset, \qquad \subsetneq, \qquad \cup, \qquad \cap \qquad \textit{and} \qquad \times$$

are

- Used
  - To denote:
    - Set membership,
    - Is a subset of a set,
    - Is a proper subset of a set,
    - Set union,
    - Set intersection,
    - Cross product
      - Respectively.

2 Isomorphism algorithm

2.1 Basic automata theory

Let:

$$G$$

be

- An undirected,
    - Unlabeled
        - ▷ And connected:
            - Graph.
- And
    - Let:

$$v_i \qquad \textit{and} \qquad v_t$$

be

- Two
    - Vertices
        - ▷ In:
            - It.
- Then
    - Randomly
        - ▷ Assign:

$$\textit{a label} \qquad \textit{and} \qquad \textit{a direction}$$

to

- All
    - Edges

      - ▷ In:
        - – It,

such that,

- There
  - ○ Will:
    - ▷ Be

at

- Least
  - ○ One
    - ▷ Path
      - – From:

v_i *to*: v_t.

- Then
  - ○ Rename
    - ▷ The vertices,
      - – Such that:

v_i *becomes*: q_i, v_t q_t, $\ldots$,

- And
  - ○ Call
    - ▷ Them:
      - – States.
- And
  - ○ Also
    - ▷ Call:

q_i

the

- Initial
    - State,
        - And:

$$q_t$$

the

- Terminal
    - State.
        - Note that:
            - In theory,

we

- Do
    - Not:
        - Have

to

- Rename
    - Vertices
        - This:
            - Way.
- But
    - We
        - Do:
            - It

for

- Convenience.

  - And so
    - Then,
      - We see that:

$$G$$

has

- Now
  - Become
    - An automaton
      - with:

$$q_i$$

as

- The only
  - Initial
    - State,
      - And:

$$q_t$$

as

- The
  - Only
    - Terminal:
      - State.
- Note that,
  - In
    - General:
      - There

can

- Be
  - Any:
    - Number

of

- Initial
  - And
    - Terminal:
      - States.
- But
  - At
    - Present:
      - There

is

- Only
  - One:

    initial-state *and* *terminal-state.*
- Also
  - Since
    - There:
      - Is

a

- Path:

$$\sigma_1, \quad \sigma_2, \quad \ldots \quad \sigma_n$$

- From:

$$q_i \qquad \textit{to}: \qquad q_t,$$

we

- Say
    - That,
        - The
            - String:

$$\text{“}\sigma_1\sigma_2\ldots\sigma_n\text{”}$$

is

- Accepted
    - By
        - That:
            - Automaton.
- The set
    - Of
        - All:
            - Strings

accepted

- By
    - An automaton
        - Is
            - Called:

“*the behavior*

of

the automaton.”

- We
  - Use:

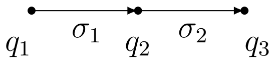
$$|\mathscr{A}|$$

to

- Denote
  - The
    - Behavior
      - Of:

$$\mathscr{A}.$$

Figure 1: Automaton $\mathscr{A}$ with q_1 as the only initial state and q_3 as the only terminal state.



- For
  - The
    - Automaton:

$$\mathscr{A}$$

in

- Figure 1:

$$|\mathscr{A}| \quad = \quad \{ \ \sigma_1\sigma_2 \ \}.$$

- Also
  - If
    - There:
      - Is

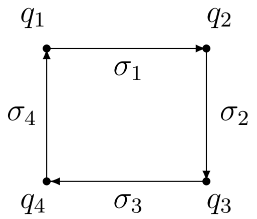
a

- Cycle
  - In
    - An:
      - Automaton,

then

- Its
  - Behavior
    - Will be:
      - Infinite.

Figure 2: Automaton $\mathscr{A}_1$ with q_1 as the only initial state and q_3 as the only terminal state.



- For

    - ○ The
        - ▷ Automaton:

$$\mathscr{A}_1$$

in

- Figure 2:

$$|\mathscr{A}_1| = \{ \sigma_1\sigma_2, \sigma_1\sigma_2\sigma_3\sigma_4\sigma_1\sigma_2, \sigma_1\sigma_2\sigma_3\sigma_4\sigma_1\sigma_2\sigma_3\sigma_4\sigma_1\sigma_2, \ldots \}.$$

- Also we see that,
    - ○ Many
        - ▷ Different:
            - Automata

can

- Be
    - ○ Generated:
        - ▷ Using

the

- Above
    - ○ Method.
        - ▷ But:
            - Even though,

the

- Automaton
    - ○ Is
        - ▷ Generated:
            - Randomly,

we see that,

- There
    - Is
        - A finite:
            - Upper bound

to

- The
    - Number
        - Of:
            - Automata

that

- Can
    - Possibly
        - Be:
            - Generated,

provided

- The
    - Set
        - Of:
            - Alphabets

used

- To
    - Label
        - The:
            - Graph

is

- Always:
    - Finite
        - And
            - Fixed.
- And so
    - When
        - We:
            - Generate

an

- Automata
    - Using
        - The above:
            - Method,

we

- Provide
    - A means
        - To:
            - Choose

a

- Particular
    - Automaton:
        - From

the

- Set

  - Of
    - All:
      - Possible automata.

- Also
  - We
    - Say:

“*an automaton is generated from*: G,”

when

- The above
  - Method
    - Is:
      - Used

to

- Generate
  - An:
    - Automaton.
- Also
  - If:

$$\mathscr{A}$$

is

- An
  - Automaton,
    - Then:

$$Q(\mathscr{A})$$

is

- The set
    - If
        - All
            - States in:

$$\mathscr{A},$$

- And:

$$I(\mathscr{A})$$

is

- The
    - Set:
        - Of

all

- Initial
    - States
        - In:

$$\mathscr{A},$$

- And:

$$T(\mathscr{A})$$

is

- The
    - Set:
        - Of

all

- Terminal
    - States
        - In:

$$\mathscr{A},$$

- And:

$$Z(\mathscr{A})$$

is

- The
    - Set:
        - Of

all

- Edges
    - In:

$$\mathscr{A}.$$

- Alternatively,
    - If:

$$\mathscr{A} \quad = \quad (Q, \quad I, \quad T, \quad Z),$$

- Then:

$$Q$$

is

- The

    - Set:
        - Of

all

- States
    - In:

$$\mathscr{A},$$

- And:

$$I$$

is

- The
    - Set:
        - Of

all

- Initial
    - States
        - In:

$$\mathscr{A},$$

- And:

$$T$$

is

- The
    - Set:
        - Of

all

- Terminal
  - States
    - In:

$$\mathscr{A},$$

- And:

$$Z$$

is

- The
  - Set:
    - Of

all

- Edges
  - In:

$$\mathscr{A}.$$

- And so
  - For
    - Any
      - Automaton:

$$\mathscr{A},$$

- We see that:

$$\begin{aligned} I(\mathscr{A}) &\subset Q(\mathscr{A}), \\ T(\mathscr{A}) &\subset Q(\mathscr{A}). \end{aligned}$$

- Also
  - If:

$$q_i \qquad \textit{and} \qquad q_j$$

are

- Two
  - States
    - ▷ Of:

$$\mathscr{A},$$

- And
  - If
    - ▷ There:
      - Is

an

- Edge
  - From:

$$q_i \qquad \textit{to}: \qquad q_j \qquad \textit{labeled}: \qquad \sigma,$$

that

- Edge
  - Is
    - ▷ Represented
      - As:

$$(q_i, \quad \sigma, \quad q_j).$$

- Also

    - If:

$$e$$

is

- An
    - Edge,
        - Then:

$$\pi_1(e)$$

is

- The
    - State
        - From
            - Which:

“e *begins*,”

- And:

$$\pi_2(e)$$

is

- The
    - Label
        - Used
            - In:

$$e,$$

- And:

$$\pi_3(e)$$

is

- The
    - State
        - In:
            - Which

$$“e \quad \textit{ends.}”$$

- Exemplifying,
    - If:

$$(q_i, \quad \sigma, \quad q_j)$$

is

- The
    - Representation
        - Of:

$$e$$

- Then:

$$\begin{aligned} \pi_1(e) &= q_i, \\ \pi_2(e) &= \sigma, \\ \pi_3(e) &= q_j. \end{aligned}$$

- Also
    - In
        - The:
            - Same

way,

- Two
  - Graphs
    - Can:
      - Have

the

- Same
  - Names:
    - For

all

- Vertices,
  - We see that,
    - Two:
      - Automata

can

- Also
  - Have
    - Same:
      - State names.
- And
  - So
    - Let:

$$e_1 \in Z(\mathscr{A}_1) \qquad \textit{and} \qquad e_1 \in Z(\mathscr{A}_2).$$

- Then
  - We
    - Say

      - That:

$$e_1 \qquad \textit{and} \qquad e_2$$

have

- The
  - Same:
    - ▷ Representation,

if

- And
  - Only
    - ▷ If:

$$\begin{aligned} \pi_1(e_1) &= \pi_1(e_2), \\ \pi_2(e_1) &= \pi_2(e_2), \\ \pi_3(e_1) &= \pi_3(e_2). \end{aligned}$$

- Exemplifying,
  - If
    - ▷ The
      - Representation of:

$$e_1$$

- Is:

$$(q_1, \quad \sigma, \quad q_2),$$

- And
  - That
    - ▷ Of:

$$e_2$$

- Is:

$$(q_1, \quad \sigma, \quad q_2),$$

- Then:

$$e_1 \qquad\qquad \textit{and} \qquad\qquad e_2$$

have

- The
  - Same:
    - Representations.
- But if
  - The
    - Representation
      - Of:

$$e_1$$

- Is:

$$(q_1, \quad \sigma, \quad q_2),$$

- And
  - That
    - Of:

$$e_2$$

- Is:

$$(q_1, \quad \sigma, \quad q_3),$$

then

- The
    - Representations
        - ▹ Of:

$$e_1 \qquad \textit{and} \qquad e_2$$

are

- Different.
    - Also:
        - ▹ If

the

- Representation
    - Of:

$$e_1$$

- Is:

$$(q_1, \quad \sigma_1, \quad q_2),$$

- And
    - That
        - ▹ Of:

$$e_2$$

- Is:

$$(q_1, \quad \sigma_2, \quad q_2),$$

then

- The
    - Representations

▹ Of:

$$e_1 \qquad\qquad \textit{and} \qquad\qquad e_2$$

are

- Different.
    - ○ Let:

$$\mathscr{A}_1 \qquad\qquad \textit{and} \qquad\qquad \mathscr{A}_2$$

be

- Two
    - ○ Automata.
        - ▹ Then
            - – We say:

$$Z(\mathscr{A}_1) \quad \subset \quad Z(\mathscr{A}_2),$$

if

- For
    - ○ All
        - ▹ Edges
            - – In:

$$Z(\mathscr{A}_1)$$

there

- Is
    - ○ An
        - ▹ Edge
            - – In:

$$Z(\mathscr{A}_2)$$

with

- The
    - Same
        - Representation.
            - Also:

$$\Sigma(\mathscr{A})$$

is

- Used
    - To:
        - Denote

the

- Set
    - Of
        - All:
            - Alphabets

used

- In
    - The
        - Edges
            - Of:

$$\mathscr{A}.$$

- Exemplifying,
    - For
        - The
            - Automaton:

$$\mathscr{A}_1$$

in

- Figure 2:

$$\Sigma(\mathscr{A}_1) \quad = \quad \{ \quad \sigma_1, \quad \sigma_2, \quad \sigma_3, \quad \sigma_4 \quad \}.$$

- Let:

$$\mathscr{A}$$

be

- An
    - Automaton,
        - And
            - Let:

$$q_i \qquad \textit{and} \qquad q_t$$

be

- The
    - Only:

initial-state *and* *terminal-state*

in

- It
    - Respectively.
        - Also
            - Let:

$$q_a \in Q(\mathscr{A}),$$

such that,

- There
    - Is
        - ▷ A path
            - From:

$$q_t \qquad \textit{to}: \qquad q_a.$$

- And
    - Also
        - ▷ Let:
            - There

be

- No
    - Path
        - ▷ From:

$$q_a \qquad \textit{to}: \qquad q_t.$$

- Then
    - It
        - ▷ Is:
            - Obvious

that,

- If:

$$q_a$$

is

- Removed
    - From:

$$\mathscr{A},$$

its

- Behavior
    - ○ Will
        - ▷ Not:
            - Change.
- And
    - ○ So
        - ▷ We:
            - Call

such

- States:

$$\text{“\textit{useless-states}.”}$$

- Let:

$$q_b \quad \in \quad Q(\mathscr{A}),$$

such that,

- There is
    - ○ No
        - ▷ Path
            - From:

$$q_i \qquad\qquad \textit{to}: \qquad\qquad q_b.$$

- Then
    - ○ It
        - ▷ Is:

            - Obvious

that,

- If:

$$q_b$$

is

- Removed
    - From:

$$\mathscr{A},$$

its

- Behavior
    - Will
        - ▹ Not:
            - Change.
- And
    - So
        - ▹ We:
            - Call

such

- States:

“*useless-states.*”

- The
    - Automaton:
        - ▹ Obtained

by

- Removing
    - All
        - Useless:
            - States

from

- Say:

$$\mathscr{A}$$

is

- Called:

$$\text{“}\textit{the trim of}\text{:}\qquad \mathscr{A}\text{.”}$$

- Briefly
    - Put,
        - We:
            - Say

that,

- An
    - Automaton
        - Is:
            - Trim,

if

- From
    - Every
        - Initial:
            - State,

there

- Is
    - A path:
        - To

at

- Least
    - One
        - Terminal:
            - State,
- And
    - All
        - Terminal:
            - States

can

- Be
    - Reached:
        - From

at

- Least
    - One
        - Initial:
            - State,
- And
    - All
        - States:

      - Can

be

- Reached
  - From
    - At:
      - Least

one

- Initial
  - State,
    - And:
      - There

is

- A path
  - From
    - All:
      - States

to

- At least
  - One
    - Terminal:
      - State.
- Also
  - We
    - Use:

$$trim(\mathscr{A})$$

to

- Denote:

“*the trim of*: $\mathscr{A}$.”

- Note that,
    - The
        - Automaton:
            - Generated

by

- The
    - Above:
        - Method

may

- Not
    - Be:
        - Trim.
- But
    - If
        - The
            - Function:

trim

is

- Applied
    - After
        - Generating:

            - An automaton

using

- The
    - Above:
        - Method,

we

- Say
    - That,
        - A trim:
            - Automaton

is

- Generated.
    - Also
        - It is:
            - A theorem

that,

- There
    - Can be
        - Two:
            - Automata,

such that,

- They
    - Both
        - Will:
            - Have

the

- Same
    - Behaviors,
        - And:
            - At

the

- Same time
    - Have
        - Different:
            - Number

of

- States.
    - And so
        - From:
            - This,
- And
    - Since
        - The:
            - Number

of

- States
    - In
        - An:
            - Automaton

cannot

- Be
  - ○ Zero,
    - ▹ We see that:
      - – For

every

- Automaton,
  - ○ There
    - ▹ Will:
      - – Be

an

- Equivalent
  - ○ Automaton
    - ▹ With:
      - – Minimal number

of

- States.
  - ○ See
    - ▹ Ullman [1]:
      - – For details.
- Let:

$$\mathscr{A}_1 \qquad \textit{and} \qquad \mathscr{A}_2$$

be

- Two
  - ○ Minimal
    - ▹ Automata,

      - Such that:

$$|\mathscr{A}_1| \quad = \quad |\mathscr{A}_2|.$$

- Then
  - ○ We see that,
    - ▷ There:
      - Exists

an

- Unique
  - ○ Mapping:
    - ▷ Between

all

- Elements
  - ○ Of:

$$Q(\mathscr{A}_1) \qquad \textit{and} \qquad Q(\mathscr{A}_2),$$

- And
  - ○ Also:
    - ▷ Between

all

- Elements
  - ○ Of:

$$Z(\mathscr{A}_1) \qquad \textit{and} \qquad Z(\mathscr{A}_2).$$

- See
  - ○ Ullman [1]

        - ▹ For:
            - Proof.

- And
    - ○ So
        - ▹ From:
            - This,

we see that,

- The
    - ○ Underlying
        - ▹ Graphs
            - Of:

$$\mathscr{A}_1 \qquad \textit{and} \qquad \mathscr{A}_2$$

will

- Be:
    - ○ Isomorphic.

Lemma 1. *Let:*

$$\mathscr{A}_1 \qquad \textit{and} \qquad \mathscr{A}_2$$

be

- *Two*
    - ○ *Minimal*
        - ▹ *Automata,*
            - *Such that:*

$$|\mathscr{A}_1| \quad = \quad |\mathscr{A}_2|.$$

- *Then*

        - *The*
            - *Underlying*
                - *Graphs of:*

$$\mathscr{A}_1 \qquad\qquad \text{and} \qquad\qquad \mathscr{A}_2$$

will

- *Be:*
    - *Isomorphic.*

2.2 Examples

In

- All
    - Figures,
        - For:
            - The sake

of

- Readability,
    - We:
        - Do

not

- Explicitly
    - Mention
        - Which:
            - State

is

- The

  - Initial:
    - ▹ State,
- And
  - Which:
    - ▹ Is

the

- Terminal
  - State.
    - ▹ But:
      - We

assume

- That,
  - If
    - ▹ There:
      - Is

no

- Incoming
  - Edge
    - ▹ To:
      - A state,

then

- It
  - Is
    - ▹ An:
      - Initial state;

- And
    - If
        - There:
            - Is

no

- Outgoing
    - Edge
        - From:
            - A state,

then

- It
    - Is
        - A terminal:
            - State.

2.2.1 First example

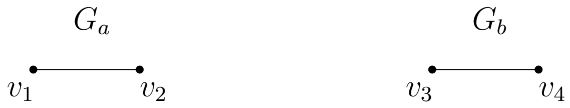
Let:

$$G_a \qquad \textit{and} \qquad G_b$$

be

- Graphs
    - As
        - Shown:
            - In figure 3.

Figure 3: Graphs G_a *and* G_b.



- Then
    - Assign:

a label *and* *a direction*

to

- The edges
    - In
        - Those:
            - Two graphs,

such that,

- The
    - Label
        - Used:
            - Will

always

- Be
    - From:

$$\Sigma = \{ \sigma \}.$$

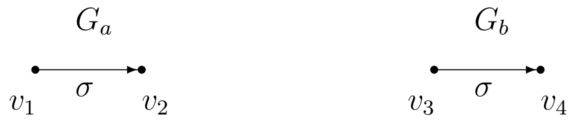
- The
    - Graphs

        - So:
            - Obtained

are

- Shown
    - In:
        - Figure 4.

Figure 4: G_b *and* G_b after directions and labels are given.



- Then
    - Rename
        - The vertices,
            - Such that:

v_1 *becomes the state*: q_1, v_2 q_2, v_3 q_3, *and* v_4 q_4.

- And
    - Call:

$$G_a$$

the

- Automaton:

$$\mathscr{A}_1$$

- And:

$$G_b$$

the

- Automaton:

$$\mathscr{A}_2.$$

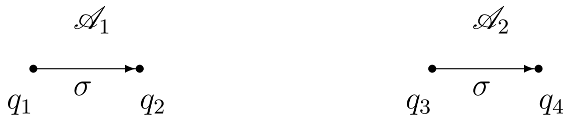
- The
  - Automata:

$$\mathscr{A}_1 \qquad \textit{and} \qquad \mathscr{A}_2$$

are

- Shown
  - In:
    - ▹ Figure 5.

Figure 5: Automata $\mathscr{A}_1$ *and* $\mathscr{A}_2$.



- Then
  - From
    - ▹ Lemma 1,
      - We see that:

$$|\mathscr{A}_1| = |\mathscr{A}_2| \qquad \textit{implies}: \qquad G_a \cong G_b.$$

- And
  - So

      - ▹ We:
        - – Use

the

- Algorithm
  - ○ To
    - ▹ Check:
      - – Whether

two

- Automata
  - ○ Have
    - ▹ The:
      - – Same behavior

to

- Develop
  - ○ The new
    - ▹ Isomorphism:
      - – Algorithm.
- Those
  - ○ Things:
    - ▹ Other

than

- Automata
  - ○ Behavior
    - ▹ Equivalence:
      - – Algorithm

that

- Are
    - ○ Required
        - ▷ To:
            - – Develop

the

- New
    - ○ Isomorphism:
        - ▷ Algorithm

is

- Illustrated
    - ○ In
        - ▷ The next:
            - – Example.

2.2.2 Second example

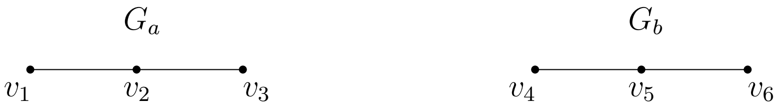
Let:

$$G_a \qquad \textit{and} \qquad G_b$$

be

- Graphs
    - ○ As
        - ▷ Shown:
            - – In figure 6.

Figure 6: Graphs G_a *and* G_b.



- Note that,
    - In
        - ▷ The previous:
            - Example,

since

- There
    - Was
        - ▷ Only:
            - One edge,

we

- Could
    - Generate:
        - ▷ Only

one

- Set
    - Of:
        - ▷ Automata

from

- Each

    - ○ Of
        - ▹ Those:
            - Graphs.

- But
    - ○ When
        - ▹ There are:
            - Three vertices,

we see that,

- Since
    - ○ Automata
        - ▹ Are:
            - Generated

at

- Random,
    - ○ Many
        - ▹ Set of:
            - Automata

can

- Be
    - ○ Generated.
        - ▹ The:
            - New algorithm

checks

- All
    - ○ Cases:
        - ▹ Until

one

- Case
    - Gives
        - A positive:
            - Result,
- Or
    - Until
        - All:
            - Cases

have

- Been
    - Exhausted.
        - We:
            - Prove

that,

- If
    - The
        - Graphs:
            - Are isomorphic

then

- At
    - Least
        - One:
            - Case

will

- Give
    - A positive:
        - Result.
- But
    - If:
        - Not;

then

- All
    - Cases
        - Will:
            - Give

a

- Negative
    - Result.
        - In this:
            - Sub sub section,

we

- Give
    - Two:
        - Cases.
- The
    - First
        - One:
            - Gives

a

- Negative
    - Result,
        - While:
            - That

of

- The
    - Second
        - Is:
            - Positive.

Case one. Randomly

- Assign:

a label *and* *a direction*

to

- The edges
    - In
        - Those:
            - Two graphs,

such that,

- The
    - Label
        - Used:
            - Will

always

- Be

- From:

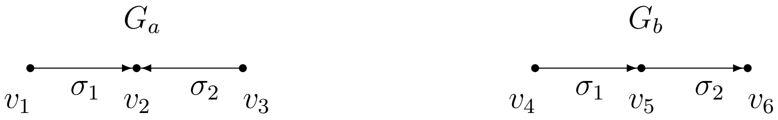
  $$\Sigma \quad = \quad \{ \ \sigma_1, \ \ \sigma_2 \ \}.$$

- The
  - Graphs
    - ▹ So:
      - – Obtained

are

- Shown
  - In:
    - ▹ Figure 7.

Figure 7: G_a *and* G_b after directions and labels are given.



- Then we see that,
  - Two
    - ▹ Trim:
      - – Automata

can

- Be
  - Generated
    - ▹ From:

$$G_a,$$

- And
    - From:

$$G_b$$

we

- Can
    - Generate:
        - Three.
- Trim
    - Automata
        - Generated
            - From:

$$G_a$$

are

- Shown
    - In:
        - Figure 8,
- And
    - That
        - From:

$$G_b$$

are

- Shown
    - In:
        - Figure 9.

Figure 8: Automata generated from G_a.

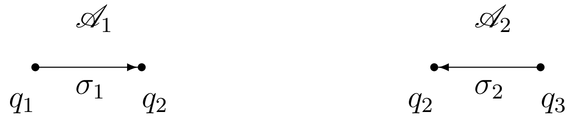
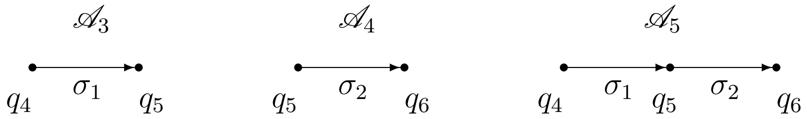


Figure 9: Automata generated from G_b.



- Then
  - We see that:

$$|\mathscr{A}_1| \quad = \quad |\mathscr{A}_3| \qquad\qquad \textit{and} \qquad\qquad |\mathscr{A}_2| \quad = \quad |\mathscr{A}_4|.$$

- And
  - So
    - Map:

$$\mathscr{A}_1 \qquad\qquad \textit{to}: \qquad\qquad \mathscr{A}_3,$$

- And:

$$\mathscr{A}_2 \qquad\qquad \textit{to}: \qquad\qquad \mathscr{A}_4.$$

- And
  - Consider
    - The
      - Pairs:

$$(\mathscr{A}_1, \quad \mathscr{A}_2) \qquad\qquad \textit{and} \qquad\qquad (\mathscr{A}_3, \quad \mathscr{A}_4).$$

- Then
    - ○ We see that,
        - ▷ The:
            - – Names

of

- The
    - ○ Terminal:
        - ▷ States

of

- Both
    - ○ Automata
        - ▷ In:

$$(\mathscr{A}_1, \quad \mathscr{A}_2)$$

are

- The
    - ○ Same.
        - ▷ But
            - – In:

$$(\mathscr{A}_3, \quad \mathscr{A}_4)$$

the

- Terminal states
    - ○ Are
        - ▷ Named:
            - – Differently.

- And so
    - We see that,
        - ▹ Even
            - Though:

$$|\mathscr{A}_1| = |\mathscr{A}_3| \qquad \textit{and} \qquad |\mathscr{A}_2| = |\mathscr{A}_4|,$$

there

- Is
    - No
        - ▹ Mapping:
            - Between

the

- Conditions
    - Satisfied
        - ▹ By:
            - The names

of

- The initial
    - And
        - ▹ Terminal:
            - States

in

- The
    - Pairs:

$$(\mathscr{A}_1, \quad \mathscr{A}_2) \qquad \textit{and} \qquad (\mathscr{A}_3, \quad \mathscr{A}_4).$$

- And

  - So:
    - Isomorphism

of

- Those
  - Two:
    - Graphs

does

- Not
  - Follow
    - In:
      - This case.

Case two. Randomly

- Assign:

a label *and* *a direction*

to

- The
  - Edges
    - In:
      - Each

of

- Those
  - Two:
    - Graphs,

such that,

- The
    - Label
        - ▷ Used:
            - Will

always

- Be
    - From:

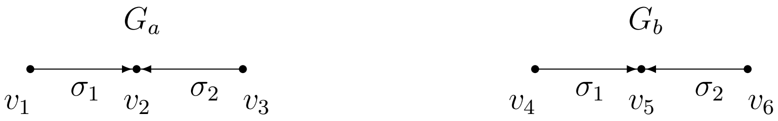
$$\Sigma \qquad = \qquad \{ \ \sigma_1, \ \ \sigma_2 \ \}.$$

- The
    - Graphs
        - ▷ So:
            - Obtained

are

- Shown
    - In:
        - ▷ Figure 10.

Figure 10: G_a *and* G_b after directions and labels are given.



- Then we see that,
    - Two
        - ▷ Trim:

      - Automata

can

- Be
  - Generated
    - From
      - Both:

$$G_a \qquad \textit{and} \qquad G_b.$$

- Trim
  - Automata
    - Generated
      - From:

$$G_a$$

are

- Shown
  - In:
    - Figure 11,
- And
  - That
    - From:

$$G_b$$

are

- Shown
  - In:
    - Figure 12.

Figure 11: Automata generated from G_a.

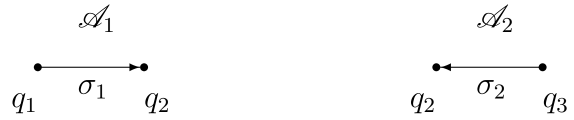
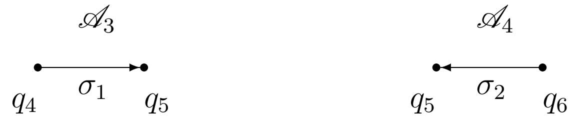


Figure 12: Automata generated from G_b.



- Then
  - ○ We see that:

$$|\mathscr{A}_1| \quad = \quad |\mathscr{A}_3| \qquad\qquad \textit{and} \qquad\qquad |\mathscr{A}_2| \quad = \quad |\mathscr{A}_4|.$$

- And
  - ○ So
    - ▷ Map:

$$\mathscr{A}_1 \qquad\qquad \textit{to}: \qquad\qquad \mathscr{A}_3,$$

- And:

$$\mathscr{A}_2 \qquad\qquad \textit{to}: \qquad\qquad \mathscr{A}_4.$$

- And
  - ○ Consider
    - ▷ The
      - – Pairs:

$$(\mathscr{A}_1, \quad \mathscr{A}_2) \qquad\qquad \textit{and} \qquad\qquad (\mathscr{A}_3, \quad \mathscr{A}_4).$$

- Then
    - ○ We see that,
        - ▷ The:
            - Names

of

- The
    - ○ Terminal:
        - ▷ States

of

- Both
    - ○ Automata
        - ▷ In:

$$(\mathscr{A}_1, \quad \mathscr{A}_2)$$

are

- The
    - ○ Same.
        - ▷ And so
            - Is in:

$$(\mathscr{A}_3, \quad \mathscr{A}_4).$$

- And
    - ○ Also
        - ▷ The:
            - Names

of

- The
    - Initial:
        - States

of

- Both
    - Automata
        - In:

$$(\mathscr{A}_1, \quad \mathscr{A}_2)$$

are

- Different,
    - And so
        - Is
            - In:

$$(\mathscr{A}_3, \quad \mathscr{A}_4).$$

- Then
    - Since:

$$|\mathscr{A}_1| \quad = \quad |\mathscr{A}_3| \qquad\qquad \textit{and} \qquad\qquad |\mathscr{A}_2| \quad = \quad |\mathscr{A}_4|,$$

- And
    - Since
        - There:
            - Is

a

- Mapping
    - Between

    - The conditions:
      - Satisfied

the

- Names of
  - Initial
    - And terminal:
      - States

in

- The
  - Pairs:

$$(\mathscr{A}_1, \quad \mathscr{A}_2) \qquad \textit{and} \qquad (\mathscr{A}_3, \quad \mathscr{A}_4),$$

we see that,

- Isomorphism of
  - Those
    - Two:
      - Graphs

follows

- In
  - This:
    - Case.

2.3 Assumptions, notations and definitions

For

- The
    - ◦ Time
        - ▷ Being,
            - – We use:

$$*$$

to

- Denote
    - ◦ Integer
        - ▷ Multiplication.
            - – Let:

$$S = \{ 10, 25 \}.$$

- And
    - ◦ Let:

$$m = 10^{th} \textit{ prime-number } * 25^{th} \textit{ prime-number.}$$

- Then
    - ◦ We see that,
        - ▷ If

we

- Know
    - ◦ That:

$$m$$

was

- Constructed
    - From:

“*a set*”

which

- Contains
    - Only:

“*integers*”

using

- The
    - Above:
        - ▷ Method,

then

- We
    - Can
        - ▷ Re-construct:

$$S$$

back

- From:

$$m.$$

- And
    - So
        - ▷ In:

      - General,

we see that,

- An
  - Unique:
    - Integer

can

- Be
  - Constructed
    - For
      - Any:

“*finite-object.*”

- And
  - Also
    - If:
      - We

are

- Given:

“*an integer,*”

- And
  - The
    - Method:
      - Used

to

- Construct

    - It,
        - Then:
            - We

can

- Re-construct
    - The
        - Original:

“*finite-object*”

back

- From
    - That:

“*integer.*”

- We call
    - That
        - Unique
            - Integer:

“*the Gödel-number*”

of

- That:

“*object.*”

- And so
    - In
        - The above
            - Example:

$$m$$

- Is:

“*the Gödel-number*”

- Of:

$$S.$$

- Also
    - Since:
        - We

are

- Free
    - To:
        - Choose

the

- Algorithm
    - To
        - Construct:

“*Gödel-numbers*,”

for

- The
    - Sake
        - Of:
            - Simplicity,

we

- Avoid
    - Complicated
        - ▷ Arguments:
            - Like

whether

- The
    - Function
        - ▷ Is:
            - Inversable,
- And
    - So:
        - ▷ On.
- And
    - So
        - ▷ We:
            - Choose

an

- Algorithm
    - That
        - ▷ Is:
            - Invertible.
- And so
    - When
        - ▷ We
            - Say:

“*a Gödel-number,*”

we

- Mean
    - ○ That,
        - ▷ The:
            - – Algorithm

to

- Construct
    - ○ It
        - ▷ Is:
            - – Given,
- And
    - ○ Also:
        - ▷ Invertible.
- Also
    - ○ We
        - ▷ Assume:
            - – That,

all

- Graphs
    - ○ Are:
        - ▷ Connected,
- And
    - ○ Does
        - ▷ Not

        - Have any:

multi-edge *or* *self-loop*,

- And
    - ○ The
        - ▷ Number of:
            - Vertices

in

- Them
    - ○ Is
        - ▷ Not
            - More than:

$$u.$$

- Also:

$$S_V$$

is

- Defined
    - ○ To
        - ▷ Be
            - The set:

$$\{ \quad v_1, \quad v_2, \quad \ldots, \quad v_u \quad \}.$$

- And
    - ○ We
        - ▷ Assume:
            - That,

the

- Set
    - ○ Of:
        - ▷ Vertices

in

- All
    - ○ Graphs
        - ▷ Whether:

directed *or* *undirected* *or* *labeled* *or* *unlabeled*

is

- A subset
    - ○ Of:

$$S_V.$$

- Also:

$$S_G$$

is

- Defined
    - ○ To
        - ▷ Be:
            - The set

of

- All
    - ○ Graphs
        - ▷ With:

$u - 1$ *vertices* *or* u *vertices.*

- We
    - ○ Had
        - ▷ Assumed:
            - That,

the

- Set
    - ○ Of
        - ▷ All:
            - Vertices

in

- A graph
    - ○ Is
        - ▷ A subset
            - Of:

$$S_V.$$

- And
    - ○ So:

$$S_G$$

will

- Be
    - ○ A finite
        - ▷ Set.
- Also

        - We
            - ▹ Assume
                - That:

$$G, \qquad G_a \qquad \textit{and} \qquad G_b$$

are

- Arbitrary
    - Graphs.
        - ▹ Also:

$$\Sigma$$

is

- Defined
    - To:
        - ▹ Be

a

- Set
    - Of:

$$\text{“}w \quad \textit{alphabets.}\text{”}$$

- Or:

$$\Sigma$$

can

- Be
    - Written
        - ▹ As:

$$\{ \quad \sigma_1, \quad \sigma_2, \quad \ldots, \quad \sigma_w \quad \}.$$

- Also:

a Σ-GRAPH

is

- Defined
    - To:
        - Be

a

- Directed
    - And
        - Labeled:
            - Graph,

such that,

- All
    - Edges
        - In:
            - It

is

- Labeled
    - Using
        - An
            - Element of:

$$\Sigma,$$

- And

    - No
        - Two edges:
            - In it

have

- The
    - Same:
        - Label.
- We
    - Had
        - Assumed:
            - That,

the

- Number
    - Of:
        - Edges

in

- All graphs
    - Is
        - Not
            - More than:

$$w,$$

- And
    - The
        - Cardinality
            - Of:

$$\Sigma$$

- Is:

$$w.$$

- And so
    - It
        - Is:
            - Possible

to

- Label
    - All
        - Edges:
            - Of

a

- Graphs
    - With
        - An arbitrary
            - Element of:

$$\Sigma,$$

such that,

- No
    - Two:
        - Edges

will

- Have

    - The
        - Same:
            - Label.

- And
    - So
        - We see that:
            - From

a

- Graph,
    - Many:

Σ-GRAPHS

can

- Be
    - Constructed.
        - But:
            - There

will

- Be
    - A finite
        - Limit:
            - To

the

- Number
    - Of
        - All:

that

- Can be
    - ○ Constructed
        - ▹ From:
            - A graph.
- And
    - ○ So:

$$S_\Sigma(G)$$

is

- Defined
    - ○ To
        - ▹ Be
            - The set:

$$\{\ \ G_d\ \ :\ \ G_d\ \ \textit{is a}\ \ \Sigma\text{-}\textsc{Graph}\ \ \textit{such that}\ \ G\ \ \textit{is its underlying graph}\ \ \}.$$

- And:

$$S_\Sigma$$

is

- Defined
    - ○ To
        - ▹ Be:

$$\bigcup_{G \in S_G} S_\Sigma(G).$$

- And so

    - For
        - ▷ Graphs,
            - Say:

$$G_1, \quad G_2 \quad \in \quad S_G,$$

the

- Sets:

$$S_\Sigma(G_1) \qquad \textit{and} \qquad S_\Sigma(G_2)$$

can

- Be
    - Considered:
        - ▷ As

two

- Partitions
    - Of:

$$S_\Sigma.$$

- And
    - So
        - ▷ Given
            - A graph:

$$G,$$

the

- Function:

$$\Upsilon$$

is

- Used
    - ○ To
        - ▷ Choose
            - – A specific:

$$\Sigma\text{-}\mathrm{GRAPH}$$

- From:

$$S_\Sigma(G).$$

- Note that,
    - ○ Since:

“*Gödel-numbers*”

are

- Unique,
    - ○ We
        - ▷ Can:
            - – Use them

to

- Choose
    - ○ A particular:

$$\Sigma\text{-}\mathrm{GRAPH}$$

- From:

$$S_\Sigma(G).$$

- And

    - So
        - ▹ Formally,
            - Let:

$$\Upsilon \quad : \quad S_G \times \mathbf{N} \quad \rightarrow \quad S_{\Sigma},$$

- And
    - Let:

$$G_d$$

be

- An
    - Element
        - ▹ Of:

$$S_{\Sigma}(G),$$

- Such that:

$$m$$

is

- Its:

“*Gödel-number.*”

- Then:

$$\Upsilon(G, \quad m) \quad = \quad G_d,$$

- And
    - Undefined
        - ▹ There:
            - Is

no

- Such
  - Element
    - In:

$$S_\Sigma(G)$$

with

- The
  - Gödel-number:

$$m.$$

- Also:

$$S_Q$$

is

- Defined
  - To
    - Be
      - The set:

$$\{ \quad q_1, \quad q_2, \quad \ldots, \quad q_u \quad \}.$$

Assumption 1. For

- Any
  - Automaton,
    - We:
      - Assume

that,

- The
    - Set
        - Of:
            - All

its

- States
    - Is
        - A subset
            - Of:

$$S_Q.$$

- And
    - So
        - From:
            - Assumption 1,

we see that,

- We
    - Do not
        - Consider:
            - Automata

with

- More
    - Than:

“u *states.*”

- Also:

$$S_{\mathscr{A}}$$

is

- Defined
  - To
    - Be:
      - The set

of

- All
  - Finite:
    - Automata,

such that,

- The
  - Number
    - Of:
      - Edges

in

- It
  - Is not
    - More
      - Than:

$$w,$$

- And
  - Each
    - Of:
      - The edges

in

- It
  - ○ Is
    - ▹ Labeled:
      - – With

an

- Element
  - ○ Of:

$$\Sigma,$$

- And
  - ○ Also
    - ▹ The:
      - – Number

of

- Edges
  - ○ Between
    - ▹ Any:
      - – Two states

is

- Not
  - ○ More
    - ▹ Than:
      - – One.
- Note that,
  - ○ Since

        - ▹ We:
            - Do

not

- Consider
    - ○ Automata
        - ▹ With
            - More than:

$$\text{“}u \quad \textit{states,}\text{”}$$

- We see that:

$$S_{\mathscr{A}}$$

is

- A finite
    - ○ Set.
        - ▹ We
            - Assume that:

$$\mathscr{A}, \qquad \mathscr{A}_1 \qquad \textit{and} \qquad \mathscr{A}_2$$

are

- Arbitrary
    - ○ Elements
        - ▹ Of:

$$S_{\mathscr{A}}$$

- Let:

$$G_d \quad \in \quad S_{\Sigma}(G),$$

- And

- Let:

$$v_i, \quad v_j \quad \in \quad G_d,$$

such that,

- There
    - Is
        - ▹ A path
            - From:

$$v_i \qquad \textit{to}: \qquad v_j.$$

- Then
    - Choose
        - ▹ An arbitrary:
            - Subgraph of:

$$G_d,$$

- Say:

$$G_{d_1},$$

such that,

- There
    - Is
        - ▹ At:
            - Least

one

- Path
    - From:

$$v_i \qquad \textit{to}: \qquad v_j \qquad \textit{in}: \qquad G_{d_1}.$$

- Note that,
    - Even
        - Though:
            - A subgraph

is

- Chosen
    - At:
        - Random,

that

- Randomness
    - Can
        - Be:
            - Circumvented

if

- We
    - Are
        - Given:

“*a Gödel-number*”

to

- Choose
    - The:
        - Subgraph.
- Or
    - We

        - Can:
            - Consider

all

- Integers
    - As:

“*Gödel-numbers*,”

- And
    - Then:
        - Iterate

the

- Process
    - Of
        - Choosing:
            - A subgraph

until

- We
    - Get
        - One:
            - With

the

- Given:

“*Gödel-number*.”

- Then
    - After

        - Choosing:

$$G_{d_1}$$

rename

- The
    - Vertices,
        - Such that:

$$v_i \quad \textit{becomes}: \quad q_i, \qquad v_j \quad q_j, \qquad v_r \quad q_r, \qquad \ldots.$$

- And
    - From
        - Then:
            - Onwards,

call

- Them
    - States.
        - And
            - So:

$$G_{d_1}$$

has

- Now
    - become
        - The
            - Automaton:

$$\mathscr{A}_1,$$

- With:

$$q_i \qquad \textit{and} \qquad q_j$$

as

- The
    - Only:

$$\textit{initial-state} \qquad \textit{and} \qquad \textit{terminal-state}$$

$$\textit{respectively.}$$

- Let:

$$\textit{trim}(\mathscr{A}_1) \quad = \quad \mathscr{A}.$$

- The
    - Function:

$$\Xi$$

is

- Used
    - To:
        - ▹ Emulate

the

- Above
    - Process.
        - ▹ Or:

$$\Xi(\Upsilon(G, \quad m), \quad v_i, \quad v_j, \quad k) \quad = \quad \mathscr{A},$$

- If:

“*the Gödel-number*”

- Of:

$$\mathscr{A}$$

is

- Equal
    - To:

$$k,$$

- And
    - Undefined:
        - For

all

- Other
    - Cases.
        - And:
            - So

we see that,

- If
    - There
        - Are
            - Paths:

$$p_1, \qquad p_2, \qquad \textit{and} \qquad p_3$$

- From:

$$v_i \qquad \textit{to}: \qquad v_j$$

- In:

$$G_d,$$

then

- For
    - ○ A particular
        - ▷ Value
            - – Of:

$$k,$$

- Only:

$$p_1, \quad \textit{and} \quad p_2$$

will

- Be
    - ○ Present
        - ▷ In:

$$G_{d_1}.$$

- And
    - ○ For
        - ▷ Another
            - – Value of:

$$k,$$

the

- Paths:

$$p_1, \quad p_2, \quad \textit{and} \quad p_3,$$

will

- Be
  - Present
    - In:

$$G_{d_1}.$$

- And
  - So:

$$\Xi$$

can

- Be
  - Used
    - To:
      - Generate

all

- Possible
  - Trim
    - Automata
      - From:

“a Σ-GRAPH.”

- Formally,
  - Let:

$$\Xi \quad : \quad S_\Sigma \times S_V \times S_V \times \mathbf{N} \quad \to \quad S_{\mathscr{A}}.$$

- And
  - Let:

$$\Upsilon(G, \quad m) \quad = \quad G_d,$$

- And:

$$G_{d_1} \quad \subset \quad G_d,$$

- Such that:

$$v_i \qquad \textit{and} \qquad v_j$$

are

- Two
    - Distinct
        - Vertices
            - Of:

$$G_{d_1},$$

- And
    - Also
        - That:
            - There

is

- At least
    - One
        - Path
            - From:

$$v_i \qquad \textit{to}: \qquad v_j$$

- In:

$$G_{d_1}.$$

- Let:

$$\mathscr{A}_1$$

be

- An
  - Automaton,
    - ▷ Such that:

$$I(\mathscr{A}_1) \quad = \quad \{ \quad q_i \quad \},$$

$$T(\mathscr{A}_1) \quad = \quad \{ \quad q_j \quad \},$$

- And:

$$q_s \quad \in \quad Q(\mathscr{A}_1),$$

if

- And
  - Only
    - ▷ If:

$$v_s \quad \in \quad V(G_{d_1}),$$

- And:

$$(q_a, \quad \sigma_h, \quad q_b) \quad \in \quad Z(\mathscr{A}_1),$$

if

- And
  - Only
    - ▷ If:

$$\langle v_a, \quad v_b, \quad \sigma_h \rangle \quad \in \quad E(G_{d_1}).$$

- And
    - Also
        - ▷ Let:

$$\textit{trim}(\mathscr{A}_1) \quad = \quad \mathscr{A}.$$

- Then:

$$\Xi(\Upsilon(G, \quad m), \quad v_i, \quad v_j, \quad k) \quad = \quad \mathscr{A},$$

- If:

“*the Gödel-number*”

- Of:

$$\mathscr{A}$$

is

- Equal
    - To:

$$k,$$

- And
    - Undefined:
        - ▷ For

all

- Other
    - Cases.
        - ▷ The
            - Function:

$$\Gamma$$

is

- Used
    - To:
        - Generate

the

- Set
    - Of
        - All:
            - Trim automata

that

- Can
    - Possibly
        - Be generated
            - From:

$$G_d \quad = \quad \Upsilon(G, \quad m)$$

- Using:

$$\Xi.$$

- Formally,
    - Let:

$$\Gamma \quad : \quad S_\Sigma \quad \rightarrow \quad \mathbf{P}(S_{\mathscr{A}}).$$

- And
    - Let:

$$G_d$$

be

- An
  - Element
    - Of:

$$S_\Sigma.$$

- Then:

$$\Gamma(G_d) \quad = \quad \{ \quad \mathscr{A} \quad : \quad \mathscr{A} \quad = \quad \Xi(G_d, \quad v_i, \quad v_j, \quad k) \quad \},$$

- Where:

$$v_i, \quad v_j \quad \in \quad V(G_d),$$

- And:

$$k \quad \in \quad \mathbf{N}.$$

- Also:

$$G^\star$$

is

- A set
  - Of:
    - Sets.
- And
  - An
    - Element
      - Of:

$$G^\star$$

is

- A set
  - Constructed:
    - From

an

- Element
  - Of:

$$S_\Sigma(G)$$

- Using:

$$\Gamma.$$

- Formally:

$$G^\star \quad = \quad \{ \quad C \quad = \quad \Gamma(G_d) \quad : \quad \textit{for all} \quad G_d \quad \in \quad S_\Sigma(G) \quad \},$$

- And:

$$S_G^\star \quad = \quad \bigcup_{G \in S_G} G^\star.$$

- We
  - Assume
    - That:

$$C$$

is

- An
  - Arbitrary
    - Element
      - Of:

$$G^\star,$$

- And:

$$C_a$$

is

- An
  - Arbitrary
    - Element
      - Of:

$$G_a^\star,$$

- And:

$$C_b$$

is

- An
  - Arbitrary
    - Element
      - Of:

$$G_b^\star.$$

- Let:

$$\mathscr{A}_1 \qquad \textit{and} \qquad \mathscr{A}_2$$

be

- Two automata
  - With
    - Exactly
      - One:

initial-state *and* *terminal-state.*

- Then
    - ◦ We
        - ▷ Write:

$$I(\mathscr{A}_1) \quad = \quad I(\mathscr{A}_2),$$

if

- The
    - ◦ Names:
        - ▷ Of

the

- Initial
    - ◦ States
        - ▷ Of:

$\mathscr{A}_1$ *and* $\mathscr{A}_2$

are

- The
    - ◦ Same.
        - ▷ And
            - – We write:

$$I(\mathscr{A}_1) \quad \neq \quad I(\mathscr{A}_2),$$

if

- The
    - ◦ Names:
        - ▷ Of

the

- Initial
    - States
        - Of:

$$\mathscr{A}_1 \qquad \textit{and} \qquad \mathscr{A}_2$$

are

- Different.
    - And
        - Similarly,
            - For:

$$\begin{aligned} T(\mathscr{A}_1) &= T(\mathscr{A}_2), \\ T(\mathscr{A}_1) &\neq T(\mathscr{A}_2), \\ I(\mathscr{A}_1) &= T(\mathscr{A}_2), \\ I(\mathscr{A}_1) &\neq T(\mathscr{A}_2). \end{aligned}$$

- And so
    - For
        - Two automata,
            - Say:

$$\mathscr{A}_1, \quad \mathscr{A}_2 \quad \in \quad C_a,$$

the

- Possible
    - Cases
        - Are,
            - Case 1:

$$
\begin{aligned}
I(\mathscr{A}_1) &= I(\mathscr{A}_2),\\
T(\mathscr{A}_1) &\neq T(\mathscr{A}_2),
\end{aligned}
$$

- Case 2:

$$
\begin{aligned}
I(\mathscr{A}_1) &\neq I(\mathscr{A}_2),\\
T(\mathscr{A}_1) &= T(\mathscr{A}_2),
\end{aligned}
$$

- Case 3:

$$
\begin{aligned}
I(\mathscr{A}_1) &= I(\mathscr{A}_2),\\
T(\mathscr{A}_1) &= T(\mathscr{A}_2),
\end{aligned}
$$

- Case 4:

$$
\begin{aligned}
I(\mathscr{A}_1) &= T(\mathscr{A}_2),\\
T(\mathscr{A}_1) &\neq I(\mathscr{A}_2),
\end{aligned}
$$

- Case 5:

$$
\begin{aligned}
I(\mathscr{A}_1) &= T(\mathscr{A}_2),\\
T(\mathscr{A}_1) &= I(\mathscr{A}_2),
\end{aligned}
$$

- Case 6:

$$
\begin{aligned}
I(\mathscr{A}_1) &\neq I(\mathscr{A}_2),\\
T(\mathscr{A}_1) &\neq T(\mathscr{A}_2),\\
I(\mathscr{A}_1) &\neq T(\mathscr{A}_2),\\
T(\mathscr{A}_1) &\neq I(\mathscr{A}_2).
\end{aligned}
$$

- And
  - So

      - ▹ The:
        - Above

six

- Cases
  - ○ Can
    - ▹ Be:
      - Defined

between

- All
  - ○ Pair
    - ▹ Of
      - Automata in:

$$C_a \quad \textit{and} \quad C_b.$$

- We
  - ○ Say:

$$C_a \subset C_b,$$

if

- For
  - ○ All
    - ▹ Automata
      - In:

$$C_a$$

there

- Is

    - An
        - Automaton
            - In:

$$C_b$$

with

- The
    - Same:
        - Behavior,
- And
    - The above
        - Mentioned:
            - Six properties

can

- Be mapped
    - From
        - All pairs
            - In:

$$C_a$$

to

- A pair
    - In:

$$C_b.$$

- Then
    - When:

$$C_a \quad \subset \quad C_b \qquad \textit{and} \qquad C_a \quad \subset \quad C_b,$$

we

- Say:

$$C_a \quad = \quad C_b.$$

- The
    - Isomorphism:
        - Algorithm

can

- Be
    - Stated
        - As:

$$C_a \quad = \quad C_b \qquad \textit{implies}: \qquad G_a \quad \cong \quad G_b.$$

- The
    - Next
        - Definition
            - Of:

$$`\subset_a'$$

defines

- The
    - First:
        - Of

the

- Above

    - ○ Six:
        - ▹ Mappings.

Definition 1. Let:

$$`\subset_a'$$

be

- A subset
    - ○ Of:

$$S^{\star}_{G} \quad \times \quad S^{\star}_{G}.$$

- Then
    - ○ We
        - ▹ Say:

$$C_a \quad \subset_a \quad C_b$$

if

- And
    - ○ Only:
        - ▹ If

for

- All:

$$\mathscr{A}_{a_1}, \quad \mathscr{A}_{a_2} \quad \in \quad C_a,$$

$$I(\mathscr{A}_{a_1}) \quad = \quad I(\mathscr{A}_{a_2}),$$

$$T(\mathscr{A}_{a_1}) \quad \neq \quad T(\mathscr{A}_{a_2})$$

implies

- That

  - There
    - ▷ Exists:

$$\mathscr{A}_{b_1},\quad \mathscr{A}_{b_2} \quad \in \quad C_b,$$

- Such that:

$$\begin{aligned} |\mathscr{A}_{a_1}| &= |\mathscr{A}_{b_1}|, \\ |\mathscr{A}_{a_2}| &= |\mathscr{A}_{b_2}|, \\ I(\mathscr{A}_{b_1}) &= I(\mathscr{A}_{b_2}), \\ T(\mathscr{A}_{b_1}) &\neq T(\mathscr{A}_{b_2}). \end{aligned}$$

- Also
  - We
    - ▷ Can
      - Define:

$\subset_b,\quad \subset_c,\quad \subset_d,\quad \subset_e\quad$ *and* $\quad \subset_f$

for

- The
  - Other
    - ▷ Five:
      - Mappings.
- But
  - Since:
    - ▷ They

are

- Very

    - Similar:
        - They

are

- Given
    - In
        - The:
            - Appendix.
- Let:

$$\text{`}\subset\text{'}$$

be

- A subset
    - Of:

$$S_G^\star \quad \times \quad S_G^\star.$$

- Then
    - We
        - Say:

$$C_a \quad \subset \quad C_b$$

if

- And
    - Only:
        - If

for

- All:

$$\mathscr{A}_{a_1}, \quad \mathscr{A}_{a_2} \quad \in \quad C_a,$$

there

- Exists:

$$\mathscr{A}_2 \quad \in \quad C_b,$$

- Such that:

$$|\mathscr{A}_1| \quad = \quad |\mathscr{A}_2|,$$

- And:

$$C_a \quad \subset_a \quad C_b,$$

$$C_a \quad \subset_b \quad C_b,$$

$$C_a \quad \subset_c \quad C_b,$$

$$C_a \quad \subset_d \quad C_b,$$

$$C_a \quad \subset_e \quad C_b,$$

$$C_a \quad \subset_f \quad C_b.$$

- And
  - We
    - Say:

$$C_a \quad = \quad C_b$$

- If:

$$C_a \quad \subset \quad C_b \qquad \textit{and} \qquad C_b \quad \subset \quad C_a.$$

- Also:

$$\widehat{\mathscr{A}}$$

is

- Defined
    - To:
        - Be

the

- Underlying
    - Graph
        - Of:

$$\mathscr{A}.$$

- We
    - Had
        - Assumed:
            - That,

the

- Set
    - Of:
        - States

in

- All
    - Automaton:
        - Is

a

- Subset
    - Of:

$$S_Q,$$

- And
    - The
        - Set:
            - Of vertices

of

- All
    - Graphs:
        - Is

a

- Subset
    - Of:

$$S_V.$$

- And so
    - We see that,
        - Even:
            - Though,

the

- States
    - Of:

$$\mathscr{A}$$

are

- Named:

$$q_1, \qquad q_2, \qquad \ldots,$$

the

- The
    - Vertices
        - Of:

$$\widehat{\mathscr{A}}$$

will

- Be
    - Named:

$$v_1, \qquad v_2, \qquad \ldots.$$

- We
    - Will
        - Soon:
            - Prove

that,

- All
    - Automata
        - In:

$$C_a$$

are

- Minimal.
    - And so
        - If:
            - We

can

- Combine
    - All
        - Automata
            - In:

$$C_a$$

into

- A single
    - Minimal
        - Automaton,
            - Say:

$$\mathscr{A}_a,$$

- And
    - Similarly,
        - All
            - Automata in:

$$C_b$$

into

- A single
    - Minimal
        - Automaton,
            - Say:

$$\mathscr{A}_b,$$

- And

- If:

$$G_a \cong \widehat{\mathscr{A}_a}, \qquad |\mathscr{A}_a| = |\mathscr{A}_b| \qquad \textit{and} \qquad \widehat{\mathscr{A}_b} \cong G_b,$$

then

- From
  - Lemma 1
    - We see that:

$$G_a \cong G_b.$$

- And
  - So
    - To:
      - Prove that,

the

- Isomorphism
  - Algorithm
    - Indeed:
      - Works,

we

- First
  - Combine
    - All
      - Automata in:

$$C_a$$

into

- A single

    - Automata,
        - Say:

$$\mathscr{A}_a,$$

such that,

- Its
    - Underlying:
        - Graph

will

- Be
    - Isomorphic
        - To:

$$G_a.$$

- But
    - We see that:

$$C_a$$

will

- Contain
    - All
        - Possible:
            - Automata

that

- Can
    - Be
        - Generated:

        - From

an

- Element
  - Of:

$$S_\Sigma(G_a).$$

- Or
  - There:
    - Maybe

two

- Automata,
  - Say:

$$\mathscr{A}_1, \quad \mathscr{A}_2 \qquad \in \qquad C_a,$$

- Such that:

$$\mathscr{A}_1$$

was

- Generated
  - From:

$$G_1,$$

- And:

$$\mathscr{A}_2$$

- From:

$$G_2,$$

- And:

$$G_1 \subset G_2 \subset G_a.$$

- And
  - So:

$$V(G_1) \subset V(G_2)$$

- And:

$$E(G_1) \subset E(G_2).$$

- And
  - So
    - ▷ When:
      - Such

a

- Condition
  - Occurs,
    - ▷ We see that:

$$Z(\mathscr{A}_1) \subset Z(\mathscr{A}_2),$$

- And
  - Also:

$$\Sigma(\mathscr{A}_1) \subset \Sigma(\mathscr{A}_2),$$

since

- Edge
  - Labels
    - ▷ Are:

      - Unique.

- But
  - Since
    - We:
      - Are

only

- Interested
  - In
    - Getting:
      - An automaton

whose

- Underlying
  - Graph
    - Is isomorphic
      - To:

$$G_a,$$

we see that,

- When we
  - Combine
    - All
      - Automata in:

$$C_a,$$

then

- We
  - Do

    - ▹ Not:
      - Have

to

- Consider
  - ○ Automata,
    - ▹ Like:

$$\mathscr{A}_1.$$

- The
  - ○ Relation:

$$\mathscr{A}_1 \quad \subsetneq_G \quad \mathscr{A}_2$$

is

- Used
  - ○ To:
    - ▹ Define

the

- Above
  - ○ Condition,
    - ▹ And
      - The function:

$$\alpha$$

is

- Used to
  - ○ Remove
    - ▹ Automata,

- Like:

$$\mathscr{A}_1 \qquad \textit{from}: \qquad C_a.$$

Definition 2. Let:

$$`\subsetneq_G'$$

be

- A subset
  - Of:

$$S_{\mathscr{A}} \quad \times \quad S_{\mathscr{A}}.$$

- Then
  - We
    - ▹ Say:

$$\mathscr{A}_1 \quad \subsetneq_G \quad \mathscr{A}_2,$$

if

- And
  - Only
    - ▹ If:

$$\Sigma(\mathscr{A}_1) \quad \subsetneq \quad \Sigma(\mathscr{A}_2).$$

- Also
  - From
    - ▹ Definition 2,
      - We see that:

$$C$$

can

- Be
    - Considered
        - As:

“*a partially-ordered-set*”

with

- Respect
    - To:

‘$\subsetneq_G$.’

- The
    - Function:

$$\alpha$$

returns

- All
    - Maximal:
        - Elements

of

- All
    - Chains:
        - In

the

- Partially
    - Ordered
        - Set:

$$C.$$

- Formally,
    - Let:

$$\alpha \quad : \quad S_G^\star \quad \to \quad \mathbf{P}(S_{\mathscr{A}}).$$

- And
    - Let:

$$C$$

be

- An
    - Element
        - ▹ Of:

$$S_G^\star$$

- And
    - Let:

$$C'$$

be

- The
    - Set:
        - ▹ Of

all

- Maximal
    - Elements:
        - ▹ Of

all

- Chains
    - In
        - The partially
            - Ordered set:

$$C$$

with

- Respect
    - To:

$$\text{`}\subsetneq_G\text{.'}$$

- Then:

$$\alpha(C) \quad = \quad C'.$$

- Let:

$$G_1$$

be

- A biconnected
    - Subgraph
        - Of:

$$G.$$

- We
    - Say
        - That:

$$G_1$$

is

- A maximal
  - ○ Biconnected
    - ▷ Subgraph
      - Of:

$$G,$$

if

- And
  - ○ Only
    - ▷ If
      - In:

$$G,$$

there

- Is no
  - ○ Biconnected
    - ▷ Supergraph
      - Of:

$$G_1.$$

- Note that,
  - ○ Since
    - ▷ The set
      - Returned by:

$$\alpha(C)$$

contains

- Only
    - Maximal:
        - Elements

of

- All
    - Chains
        - In:

$$C,$$

we see that,

- The
    - Underlying:
        - Graphs

of

- All
    - Elements
        - In:

$$\alpha(C)$$

will

- Be maximal
    - Biconnected
        - Subgraphs
            - Of:

$$G.$$

- The

    - Operator:

$$\uplus$$

is

- Used
    - To:
        - Combine

all

- Automata
    - In:

$$C$$

into

- A single
    - Automata
        - After
            - Invoking:

$$\alpha.$$

- Formally:

$$\mathscr{A}_1 \quad \uplus \quad \mathscr{A}_2$$

is

- Defined
    - To
        - Be:

$$\mathscr{A} \quad = \quad (Q, \quad I, \quad T, \quad Z),$$

- Where:

$$\begin{array}{lcccc} Q & = & Q(\mathscr{A}_1) & \cup & Q(\mathscr{A}_2), \\ I & = & I(\mathscr{A}_1) & \cup & I(\mathscr{A}_2), \\ T & = & T(\mathscr{A}_1) & \cup & T(\mathscr{A}_2), \\ Z & = & Z(\mathscr{A}_1) & \cup & Z(\mathscr{A}_2). \end{array}$$

- Let:

$$C_a \quad = \quad \{ \quad \mathscr{A}_1, \quad \mathscr{A}_2, \quad \ldots \quad \}.$$

- Then
  - Since:

$$\Gamma$$

generates

- All
  - Possible:
    - ▹ Automata

from

- The
  - Given:

$$\Sigma\text{-}\textsc{Graph},$$

we see that,

- All
  - Elements
    - ▹ Of:

$$Q,$$

- Or almost
    - All
        - Elements
            - Of:

$$Q$$

will

- Be:

an initial-state *and* *a terminal-state*

of

- Some
    - Automata
        - In:

$$C_a.$$

- And
    - So
        - If:

$$Q = Q(\mathscr{A}_1) \cup Q(\mathscr{A}_2) \cup \ldots,$$

- And
    - If:

$$\mathscr{A}_a = \mathscr{A}_1 \uplus \mathscr{A}_2 \uplus \ldots,$$

then

- All

        - States
            - Of:

$$\mathscr{A}_a,$$

- Or almost
    - All
        - States
            - Of:

$$\mathscr{A}_a$$

will

- Be:

an initial-state *and* *a terminal-state*

- Of:

$$\mathscr{A}_a.$$

- And so
    - If
        - We
            - Use:

$$\uplus$$

to

- Combine
    - All
        - Automata
            - In:

$$C_a$$

before

- Applying:

$$\alpha,$$

then

- The
    - Resulting:
        - ▷ Automaton

will

- Not
    - Be:
        - ▷ Minimal.
- And
    - So
        - ▷ Let:

$$\alpha(C_a) \quad = \quad \{ \quad \mathscr{A}_1, \quad \mathscr{A}_2, \quad \ldots \quad \}$$

- And
    - Let:

$$\mathscr{A}_a \quad = \quad \mathscr{A}_1 \quad \uplus \quad \mathscr{A}_2 \quad \uplus \quad \ldots.$$

- Then
    - We see that,
        - ▷ For
            - All:

$$\mathscr{A}_i \ \in \ \alpha(C_a),$$

the

- Underlying
    - Graphs
        - Of:

$$\mathscr{A}$$

will

- Be
    - A maximal
        - Biconnected
            - Subgraph of:

$$G_a.$$

- And so
    - When
        - We
            - Construct:

$$\mathscr{A}_a$$

from

- The
    - Elements
        - Of:

$$\alpha(C_a)$$

- Using:

$$\uplus,$$

- We see that:

$$\mathscr{A}_a$$

will

- Have
  - States:
    - ▹ Which

are

- Neither:
  - Initial
    - ▹ Nor
      - – Terminal.
- And
  - So
    - ▹ By:
      - – That,

we

- Take
  - The
    - ▹ First:
      - – Step

to

- Construct
  - A minimal:
    - ▹ Automata.
- And

- So
  - Assume
    - That:

$$\mathscr{A}_a$$

is

- Minimal.
  - And
    - Also
      - That:

$$\widehat{\mathscr{A}_a}$$

is

- Isomorphic
  - To:

$$G_a.$$

- Then
  - After
    - We:
      - Combining

the

- Elements
  - Of:

$$C_a$$

- Using:

$$\uplus,$$

we see that,

- There
    - May
        - Be:
            - More

than

- One
    - Initial
        - And terminal:
            - State

in

- The
    - Resulting:
        - Automaton.
- And
    - So
        - Let:

$$q_{a_1} \qquad \textit{and} \qquad q_{a_2}$$

be

- Two
    - Initial
        - States
            - Of:

$$\mathscr{A}_a.$$

- Also
    - ○ Assume
        - ▷ That:

$$q_{a_1} \qquad \textit{and} \qquad q_{a_2}$$

belongs

- To:

“*a cycle.*”

- Then
    - ○ If:

$$q_{a_2}$$

is

- Removed
    - ○ From:

$$I(\mathscr{A}_a),$$

- We see that:

$$\mathscr{A}_a$$

will

- Still
    - ○ Remain
        - ▷ Minimal,
            - – And:

$$\widehat{\mathscr{A}_a}$$

will

- Still
  - Be
    - Isomorphic
      - To:

$$G_a,$$

- And
  - Only:

$$|\mathscr{A}_a|$$

will

- Change.
  - But
    - The
      - Change in:

$$|\mathscr{A}_a|$$

does

- Not
  - Concern:
    - Us,

since

- We
  - Are
    - Not:
      - Interested

in

- Any
    - Particular:
        - ▹ Behavior,
- And
    - We
        - ▹ Only
            - Want:

$$\widehat{\mathscr{A}_a} \quad\cong\quad G_a.$$

- And
    - So
        - ▹ We:
            - Remove

some

- States
    - From:

$$I(\mathscr{A}_a),$$

so that,

- No
    - Two
        - ▹ Elements
            - Of:

$$I(\mathscr{A}_a)$$

will

- Belong

    - To:

“*a cycle.*”

- We
    - Will
        - Later:
            - Explain

why

- We
    - Do:
        - This.
- Let:

$$\mathscr{A}_b$$

be

- The
    - Automaton:
        - Obtained

by

- Combining
    - All
        - Automata
            - In:

$$\alpha(C_b)$$

- Using:

$$\uplus.$$

- And
  - Let:

$$|\mathscr{A}_a| \quad = \quad |\mathscr{A}_b|.$$

- Then
  - From
    - Lemma 1,
      - We see that:

$$\widehat{\mathscr{A}_a} \quad \cong \quad \widehat{\mathscr{A}_b}.$$

- And
  - So
    - If:

$$q_{a_1} \qquad \textit{and} \qquad q_{a_2}$$

are

- Two
  - Initial
    - States
      - Of:

$$\mathscr{A}_a,$$

- And
  - If:

$$q_{a_1} \qquad \textit{and} \qquad q_{a_2}$$

belongs

- To:

“*a cycle,*”

then

- There
    - ○ Will:
        - ▷ Be

two

- Initial
    - ○ States,
        - ▷ Say:

$$q_{b_1}, \qquad \textit{and} \qquad q_{b_2}$$

- In:

$$\mathscr{A}_b,$$

such that,

- There
    - ○ Will:
        - ▷ Be

a

- Mapping
    - ○ Between:

$$q_{a_1} \qquad \textit{and} \qquad q_{b_1},$$

- And
    - ○ Also
        - ▷ Between:

$$q_{a_2} \qquad \textit{and} \qquad q_{b_2}$$

- And
  - Also:

$$q_{b_1} \qquad \textit{and} \qquad q_{b_2}$$

will

- Belong
  - To:

a cycle."

- And so
  - If
    - ▷ We
      - Remove:

$$\begin{aligned} q_{a_2} &\qquad \textit{from}: \qquad I(\mathscr{A}_a), \\ q_{b_2} &\qquad \textit{from}: \qquad I(\mathscr{A}_b), \end{aligned}$$

then

- Also:

$$|\mathscr{A}_a| \quad = \quad |\mathscr{A}_b|.$$

- But
  - If
    - ▷ We
      - Remove:

$$\begin{aligned} q_{a_2} &\qquad \textit{from}: \qquad I(\mathscr{A}_a), \\ q_{b_1} &\qquad \textit{from}: \qquad I(\mathscr{A}_b), \end{aligned}$$

- Then:

$$|\mathscr{A}_a| \quad \neq \quad |\mathscr{A}_b|.$$

- Therefore
    - Since
        - We:
            - Use

the

- Algorithm
    - To
        - Check:
            - Whether

two

- Automata
    - Have
        - The:
            - Same behavior,

we see that,

- When
    - We:
        - Remove

some

- States
    - From:

$$I(\mathscr{A}_a) \qquad \textit{and} \qquad I(\mathscr{A}_b),$$

we

- Must
    - Ensure:
        - That,

the

- Resulting
    - Behaviors
        - Of:
            - Those

two

- Automata
    - Are
        - Still:
            - The same.
- Then
    - Since
        - The:
            - Labels

on

- All
    - Edges
        - Of:

$$\mathscr{A}_a$$

are

- Unique,

    - We see that,
        - The:
            - Set

of

- Labels
    - Used:
        - In

the

- Out
    - Going
        - Edges:
            - Of

a

- State
    - Can
        - Be:
            - Used

to

- Construct:

“*a Gödel-number*”

- For:

“*a state.*”

- And
    - Also

- ▹ Since:

$$|\mathscr{A}_a| \quad = \quad |\mathscr{A}_b|,$$

- We see that:

“*the Gödel-numbers*”

for

- Each of
  - ◦ The
    - ▹ Corresponding
      - States in:

$\mathscr{A}_a$ *and* $\mathscr{A}_b$

will

- Be
  - ◦ The:
    - ▹ Same.
- And so
  - ◦ If
    - ▹ We
      - Assign:

“*a Gödel-number*”

for

- Each
  - ◦ Initial:
    - ▹ States

in

- Those
    - Two:
        - Automata,

then

- We
    - Can
        - Use:
            - It

to

- Remove
    - Elements
        - From:

$$I(\mathscr{A}_a) \qquad\qquad \textit{and} \qquad\qquad I(\mathscr{A}_b),$$

such that,

- The
    - Resulting:
        - Automata

will

- Still
    - Have
        - The:
            - Same behavior.
- The
    - Function:

$$\beta$$

returns

- The set
    - Of
        - All:
            - Alphabets

on

- The
    - Out going
        - Of:

“*a state*,”

so that,

- We
    - Can
        - Use:
            - It

to

- Assign:

“*a Gödel-number*”

to

- All
    - States
        - In:

$$I(\mathscr{A}_a).$$

- The
    - Function:

$$\delta$$

is

- Used
    - To
        - Choose:

“*a state*”

- From:

“*a cycle*”

of

- Initial
    - States,
        - Such that:

“*the Gödel-number*”

of

- The
    - Chosen
        - State:
            - Will

be

- Minimal
    - Among

      - ▷ All:

“states”

in

- That:

“cycle.”

- Assume
  - That:

$$q_{a_1}, \qquad q_{a_2} \qquad \textit{and} \qquad q_{a_3}$$

are

- Three
  - Elements
    - ▷ Of:

$$I(\mathscr{A}_a).$$

- Also
  - Assume
    - ▷ That:

$$q_{a_1} \qquad \textit{and} \qquad q_{a_2}$$

belongs

- To:

“a cycle,”

- And
  - Also

        - ▹ That:
            - There

is

- No
    - ○ Path
        - ▹ From:

$$q_{a_1} \qquad \textit{to}: \qquad q_{a_3}.$$

- Then
    - ○ If
        - ▹ We
            - Remove:

$$q_{a_3} \qquad \textit{from}: \qquad I(\mathscr{A}_a),$$

we see that,

- Some
    - ○ States:
        - ▹ Cannot

be

- Reached
    - ○ From:

$$q_{a_1}.$$

- And
    - ○ So:

$$\mathscr{A}_a$$

will

- Cease
    - To
        - Be:
            - Trim.
- And
    - So:
        - When

we

- Remove
    - Some
        - Elements
            - From:

$$I(\mathscr{A}_a),$$

we

- Partition:

$$I(\mathscr{A}_a)$$

in

- Such
    - A way
        - That:

two

- Elements
    - Of:

$$I(\mathscr{A}_a)$$

will

- Belong to
    - The
        - ▹ Same:
            - Partition,

if

- And
    - Only:
        - ▹ If

they

- Belong
    - To:

“a cycle.”

- And
    - So
        - ▹ From:

“a partition,”

if

- We
    - Remove
        - ▹ All:

“states,”

except

- For
    - ○ One,
        - ▷ Then:
            - – The automaton

will

- Still
    - ○ Remain:
        - ▷ Minimal.
- And so
    - ○ Except
        - ▷ For:
            - – One,

we

- Remove
    - ○ All
        - ▷ States
            - – From:

“*a partition*.”

- Formally,
    - ○ Let:

$$\beta \quad : \quad S_{\mathscr{A}} \times S_Q \quad \to \quad \mathbf{N}.$$

- Then:

$$\beta(\mathscr{A}, \quad q_i)$$

is

- Defined
  - To
    - ▹ Be:

"*the Gödel-number*"

- Of:

$$\{ \quad \sigma \quad : \quad (q_i, \quad \sigma, \quad q_j) \quad \in \quad Z(\mathscr{A}) \quad \textit{for all} \quad q_j \quad \in \quad Q(\mathscr{A}) \quad \}.$$

- Also
  - Let:

$$\delta \quad : \quad S_{\mathscr{A}} \quad \times \quad S_Q \quad \to \quad S_Q.$$

- And
  - Let:

$$q_i \quad \in \quad I(\mathscr{A}),$$

- And
  - Let:

$$S_{q_i}$$

be

- The
  - Set:

$$\{ \quad q_j \quad : \quad \textit{there is a path from} \quad q_i \quad \textit{to} \quad q_j$$

$$\textit{and}$$

$$\textit{a path from} \quad q_j \quad \textit{to} \quad q_i \quad \textit{in} \quad \mathscr{A}$$

$$\textit{and}$$

$$q_j \in I(\mathscr{A}) \quad \}.$$

- And
  - Let:

$$n_{q_k}$$

be

- The
  - Minimal
    - Value
      - In:

$$\{ \ n_{q_j} \ : \ n_{q_j} \ = \ \beta(\mathscr{A}, \ q_j) \quad \textit{and} \quad q_j \ \in \ (S_{q_i} \ \cup \ \{ \ q_i \ \}) \ \}.$$

- And
  - Let:

$$q_k \quad \in \quad (S_{q_i} \ \cup \ \{ q_i \}),$$

be

- Such that:

$$\beta(\mathscr{A}, \ q_k) \quad = \quad n_{q_k}.$$

- Then:

$$\delta(\mathscr{A}, \ q_i) \quad = \quad q_k,$$

- If:

$$q_i \quad \in \quad I(\mathscr{A}),$$

- And
    - Undefined:
        - For

all

- Other
    - Cases.
        - And
            - Similarly:

γ *and* ζ

are

- Used
    - To:
        - Choose

a

- Particular
    - Terminal
        - State
            - From:

“*a cycle*”

of

- Some
    - Terminal:

        - ▹ States.
- Formally,
    - ○ Let:

$$\gamma \quad : \quad S_{\mathscr{A}} \times S_Q \quad \to \quad \mathbf{N}.$$

- Then:

$$\gamma(\mathscr{A}, \quad q_i)$$

is

- Defined
    - ○ To
        - ▹ Be:

"*the Gödel-number*"

- Of:

$$\{ \ \sigma \quad : \quad (q_j, \quad \sigma, \quad q_i) \quad \in \quad Z(\mathscr{A}) \quad \textit{for all} \quad q_j \quad \in \quad Q(\mathscr{A}) \ \}.$$

- Let:

$$\zeta \quad : \quad S_{\mathscr{A}} \times S_Q \quad \to \quad S_Q.$$

- And
    - ○ Let:

$$q_t \quad \in \quad T(\mathscr{A}),$$

- And
    - ○ Let:

$$S_{q_t}$$

be

- The
  - Set:

$$\{ \quad q_j \quad : \quad \textit{there is a path from} \quad q_t \quad \text{to} \quad q_j \quad \textit{and} \quad \textit{a path from} \quad q_j \quad \textit{to} \quad q_t \quad \textit{in} \quad \mathscr{A} \quad \textit{and} \quad q_j \quad \in \quad T(\mathscr{A}) \quad \}.$$

- And
  - Let:

$$n_{q_k}$$

be

- The
  - Maximal
    - Value
      - In:

$$\{ \ n_{q_j} \ : \ n_{q_j} \ = \ \gamma(\mathscr{A}, \ q_j) \quad \textit{and} \quad q_j \quad \in \quad (S_{q_t} \ \cup \ \{ \ q_t \ \}) \ \}.$$

- And
  - Let:

$$q_k \quad \in \quad (S_{q_t} \ \cup \ \{ \ q_t \ \}),$$

be

- Such that:

$$\gamma(\mathscr{A}, \ q_k) \quad = \quad n_{q_k}.$$

- Then:

$$\zeta(\mathscr{A}, \quad q_t) \qquad = \qquad q_k,$$

- If:

$$q_t \quad \in \quad T(\mathscr{A}),$$

- And
    - ○ Undefined:
        - ▷ For

all

- Other
    - ○ Cases.
        - ▷ But:
            - It

maybe

- That
    - ○ After
        - ▷ Applying:

$$\beta \qquad\qquad \textit{and} \qquad\qquad \delta,$$

an

- Initial
    - ○ State
        - ▷ Can:
            - Still

be

- Reached

        - From
            - Another:
                - Initial-state.

- Then
    - Since
        - We:
            - Are

not

- Interested
    - In
        - Any:
            - Particular behavior,

we see that,

- If
    - An
        - Initial:
            - State

can

- Be reached
    - From
        - Another:
            - Initial-state,

then

- It
    - Can
        - Be:

      - Removed

from

- The
  - ○ Set
    - ▷ Of:
      - Initial-states.
- And
  - ○ So
    - ▷ After
      - Applying:

$$\beta \qquad \textit{and} \qquad \delta,$$

the

- Function:

$$\eta$$

is

- Used to
  - ○ Isolate
    - ▷ Initial:
      - States

that

- Cannot
  - ○ Be:
    - ▷ Reached

from

- Other
    - Initial:
        - States.
- We
    - Will
        - Later:
            - Explain

why

- We
    - Do:
        - This.
- Formally,
    - Let:

$$\eta \quad : \quad S_{\mathscr{A}} \quad \to \quad \mathbf{P}(S_Q).$$

- Then:

$$\eta(\mathscr{A})$$

is

- Defined
    - To
        - Be:

$$\{ \quad q_i \in I(\mathscr{A}) \quad : \quad \textit{there is no path} \quad \textit{from any other} \quad \textit{element of} \quad I(\mathscr{A}) \quad \textit{to} \quad q_i \quad \}.$$

- Also
    - Similarly:

$$\theta$$

is

- Used to
    - Isolate
        - Terminal:
            - States,

such that,

- Those
    - Isolated
        - Terminal:
            - States

cannot

- Be reached
    - From
        - Other:
            - Terminal-states.
- Formally,
    - Let:

$$\theta \quad : \quad S_{\mathscr{A}} \quad \to \quad \mathbf{P}(S_Q).$$

- Then:

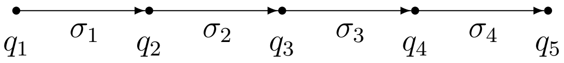
$$\theta(\mathscr{A})$$

is

- Defined
  - To
    - ▹ Be:

$$\{ \quad q_t \in T(\mathscr{A}) \quad : \quad \textit{there is no path from } q_t \textit{ to any other element of } T(\mathscr{A}) \}.$$

Figure 13: Automata $\mathscr{A}$ with $I(\mathscr{A}) = \{ q_1, \ q_2, \ q_3, \ q_4 \}, \quad T(\mathscr{A}) = \{ q_2, \ q_3, \ q_4, \ q_5 \}$



- For
  - The
    - ▹ Automaton:

$$\mathscr{A}$$

in

- Figure 13:

$$\eta(\mathscr{A}) = \{ q_1 \} \quad \textit{and} \quad \theta(\mathscr{A}) = \{ q_5 \}.$$

- The
  - Function:

$$\kappa$$

is

- The
    - ○ Place
        - ▷ Where:

δ, ζ, η *and* θ

are

- Used to
    - ○ Remove
        - ▷ Some:
            - States

from

- The
    - ○ Set
        - ▷ Of:

initial-states *and* *terminal-states.*

- The
    - ○ Function:

$$\kappa$$

initially

- Uses:

δ *and* ζ

to

- Construct

    - The
        - Automaton:

$$\mathscr{A}_2,$$

- And
    - Then
        - Use:

$$\eta \qquad \textit{and} \qquad \theta$$

to

- Remove
    - Some:
        - States

from

- The
    - Set
        - Of:

initial-states *and* *terminal-states*

- In:

$$\mathscr{A}_2$$

so as

- To
    - Construct:

$$\mathscr{A}.$$

- Formally,

- Let:

$$\kappa \quad : \quad S_{\mathscr{A}} \quad \rightarrow \quad S_{\mathscr{A}}.$$

- And
  - Let:

$$\mathscr{A}_1 \quad \in \quad S_{\mathscr{A}},$$

- And
  - Let:

$$I_2 \quad = \quad \{ \quad q_{i2} \quad = \quad \delta(\mathscr{A}_1, \quad q_i) \quad : \quad \textit{for all} \quad q_i \quad \in \quad I(\mathscr{A}_1) \quad \},$$

$$T_2 \quad = \quad \{ \quad q_{t2} \quad = \quad \zeta(\mathscr{A}_1, \quad q_t) \quad : \quad \textit{for all} \quad q_t \quad \in \quad T(\mathscr{A}_1) \quad \}.$$

- And
  - Let:

$$\mathscr{A}_2 \quad = \quad (Q(\mathscr{A}_1), \quad I_2, \quad T_2, \quad Z(\mathscr{A}_1)),$$

$$\mathscr{A} \quad = \quad (Q(\mathscr{A}_2), \quad \eta(\mathscr{A}_2), \quad \theta(\mathscr{A}_2), \quad Z(\mathscr{A}_2)).$$

- Then:

$$\kappa(\mathscr{A}_1) \quad = \quad \mathscr{A}.$$

- Let:

$$C$$

be

- An
  - Element
    - Of:

$$S^{\star}_{G}.$$

- The
    - Function:

$$\Theta$$

is

- Used to
    - Combine
        - All
            - Automata in:

$$C$$

into

- A single
    - Minimal:
        - Automaton.
- For
    - That,
        - It

will

- First
    - Use:

$$\alpha \quad \textit{and} \quad \uplus$$

to

- Combine
    - All

- ▷ Automata
  - In:

$$C.$$

- And
  - Then
    - ▷ Use:

$$\kappa$$

to

- Get
  - The final
    - ▷ Minimal:
      - Automaton.
- Formally,
  - Let:

$$\Theta \quad : \quad \mathbf{P}(S_{\mathscr{A}}) \quad \to \quad S_{\mathscr{A}}.$$

- And
  - Let:

$$\begin{aligned} \alpha(C) &= \{ \mathscr{A}_1, \mathscr{A}_2, \ldots, \mathscr{A}_n \}, \\ \mathscr{A} &= \mathscr{A}_1 \uplus \mathscr{A}_2 \uplus \ldots \uplus \mathscr{A}_n. \end{aligned}$$

- Then:

$$\Theta(C) = \kappa(\mathscr{A}).$$

2.4 Proving the algorithm

Let:

$$\Xi(\Upsilon(G, \quad m), \quad v_i, \quad v_j, \quad k) \quad = \quad \mathscr{A}.$$

- And
    - Let:

$$\mathscr{A}_d$$

be

- An
    - Automaton
        - Obtained:
            - By removing

an

- Arbitrary
    - Edge,
        - Say:

$$(q_a, \quad \sigma, \quad q_b)$$

- From:

$$Z(\mathscr{A}).$$

- Then
    - Since:

$$\sigma$$

is

- Used

        - Only:
            - In

one

- Edge
    - Of:

$$\mathscr{A},$$

- And
    - Since:

$$\mathscr{A}$$

is

- Trim,
    - We see that,
        - There:
            - Will exist

a

- String:

$$\mathrm{S},$$

such that,

- It
    - Will
        - Contain
            - The alphabet:

$$\sigma,$$

- And:

$$\mathrm{S} \quad \in \quad |\mathscr{A}| \qquad\qquad \textit{and} \qquad\qquad \mathrm{S} \quad \notin \quad |\mathscr{A}_d|.$$

- And
    - So:

$$\mathscr{A}$$

will

- Be minimal.
    - Note that,
        - The resulting
            - Automaton of:

$$\Xi(\Upsilon(G,\ m),\ v_i,\ v_j,\ k)$$

is

- Forcefully
    - Made:
        - Minimal,

so that,

- It
    - Will
        - Be:
            - Possible

to

- Get
    - The
        - Original:

            - Subgraph

from

- Which
    - The
        - ▹ Automaton:
            - Was generated.

Lemma 2. *Let:*

$$\Xi(\Upsilon(G,\ m),\ v_i,\ v_j,\ k) \ = \ \mathscr{A}.$$

- *Then:*

$$\mathscr{A}$$

- *Is:*

"minimal."

Notation 1. When

- We
    - Say:

"There is a mapping between

the boundary conditions

among all

automata in: C_a *and* C_b,"

we

- Mean:

$$C_a \subset_a C_b$$

$$C_a \subset_b C_b$$

$$C_a \subset_c C_b$$

$$C_a \subset_d C_b$$

$$C_a \subset_e C_b$$

$$C_a \subset_f C_b$$

and

vice-versa.

- Assume:

$$C_a = C_b.$$

- Also
  - Assume
    - That:

C_a *and* C_b

was

- Constructed
  - From
    - The
      - Σ-GRAPHS:

G'_a *and* G'_b

respectively.

- Let:

$$\mathscr{A}_1 \quad \in \quad C_a,$$

such that,

- It
    - Is
        - ▹ A maximal:
            - Element

in

- The
    - Partially
        - ▹ Ordered
            - Set:

$$C_a.$$

- Also
    - Since:

$$C_a \quad = \quad C_b,$$

- Let:

$$\mathscr{A}_3$$

be

- An
    - Element
        - ▹ Of:

$$C_b,$$

- Such that:

$$|\mathscr{A}_1| \quad = \quad |\mathscr{A}_3|.$$

- Also
    - From
        - This,
            - We see that:

$$\Sigma(\mathscr{A}_1) \quad = \quad \Sigma(\mathscr{A}_3). \tag{1}$$

- Let:

$$\mathscr{A}_6$$

be

- An
    - Element
        - Of:

$$C_b,$$

- Such that:

$$\Sigma(\mathscr{A}_3) \quad \subsetneq \quad \Sigma(\mathscr{A}_6).$$

- Then
    - Since
        - No two:
            - Edges in:

$$G'_a,$$

have

- The
    - Same:

      - ▷ Label,
- And
    - ○ Since:

$$C_a \quad = \quad C_b,$$

we see that,

- There
    - ○ Will:
        - ▷ Exist

an

- Automaton,
    - ○ Say:

$$\mathscr{A}_5$$

- In:

$$C_a,$$

- Such that:

$$\Sigma(\mathscr{A}_1) \quad \subsetneq \quad \Sigma(\mathscr{A}_5),$$

which

- Inturn
    - ○ Will:
        - ▷ Contradict

the

- Fact
    - ○ That:

$$\mathscr{A}_1$$

is

- A maximal
    - Element
        - ▹ And:
            - So

we see that,

- There
    - Will
        - ▹ Not:
            - Exist

an

- Automaton,
    - Say:

$$\mathscr{A}_6,$$

- Such that:

$$\Sigma(\mathscr{A}_3) \quad \subsetneq \quad \Sigma(\mathscr{A}_6).$$

- And
    - So:

$$\mathscr{A}_3$$

will

- Also
    - Be

        - A maximal:
            - Element

in

- The
    - Partially
        - Ordered
            - Set:

$$C_b.$$

- Let:

$$\mathscr{A}_3$$

be

- An
    - Element
        - Of:

$$C_a,$$

- Such that:

$$\mathscr{A}_2 \quad \subsetneq_G \quad \mathscr{A}_1. \tag{2}$$

- Then
    - Since:

$$C_a \quad = \quad C_b,$$

we see that,

- There
    - Will:

        - ▷ Exist

an

- Automaton,
    - ○ Say:

$$\mathscr{A}_4 \quad \in \quad C_b,$$

- Such that:

$$|\mathscr{A}_2| \quad = \quad |\mathscr{A}_4|. \tag{3}$$

- Then
    - ○ From:
        - ▷ Statements 1 and 2,
- And
    - ○ Since:

$$C_b$$

was

- Constructed
    - ○ From:

“*a*   Σ-GRAPH,”

- We see that:

$$\mathscr{A}_4 \quad \subsetneq_G \quad \mathscr{A}_3. \tag{4}$$

- And
    - ○ So
        - ▷ From:
            - Statements 1, 2, 3 and 4,

we see that,

- After
    - Performing:

$$\alpha(C_a) \qquad \textit{and} \qquad \alpha(C_b),$$

the

- Automata:

$$\mathscr{A}_2 \qquad \textit{and} \qquad \mathscr{A}_4$$

will

- Be
    - Removed
        - From:

$$C_a \qquad \textit{and} \qquad C_b$$

respectively.

- But:

$$\mathscr{A}_1 \qquad \textit{and} \qquad \mathscr{A}_3$$

will

- Not
    - Be
        - Removed
            - From:

$$C_a \qquad \textit{and} \qquad C_b,$$

since

- They

  - ○ Both
    - ▹ Are maximal:
      - – Elements

in

- Their
  - ○ Respective
    - ▹ Partially ordered:
      - – Sets.
- And
  - ○ So:

$$\alpha(C_a) \quad = \quad \alpha(C_b)$$

will

- Still
  - ○ Hold
    - ▹ When:

$$C_a \quad = \quad C_b.$$

Lemma 3. *Let:*

$$C_a \quad = \quad C_b.$$

- *Then:*

$$\alpha(C_a) \quad = \quad \alpha(C_b).$$

- Let:

$$\alpha(C) \quad = \quad C'.$$

- And

  - Let:

$$G_1$$

be

- A maximal
  - Biconnected
    - Subgraph
      - Of:

$$G.$$

- Then
  - Since:

$$C$$

contains

- All
  - Possible:
    - Automata

that

- Can
  - Be
    - Generated:
      - From

an

- Element
  - Of:

$$S_G^\star$$

we see that,

- There
    - May
        - Be:
            - Many automata
- Or
    - At least
        - One
            - In:

$$C'$$

- Such that:

$$G_1$$

is

- The
    - Underlying
        - Graph:
            - Of

all

- Of
    - Them.
        - Note that,
            - If:

$$G_d$$

is

- A maximal
    - Biconnected
        - Subgraph
            - Of:

“*a* Σ-Graph,”

- And if
    - All
        - Vertices
            - Of:

$$G_d$$

can

- Be
    - Reached
        - From:
            - All

the

- Other
    - Vertices,
        - Then many:
            - Trim-automata

could

- Be
    - Generated
        - From:

$$G_d,$$

such that,

- Except
  - For
    - Their:

initial-state *and* *terminal-state*,

everything

- Else in
  - All
    - Those:
      - Automata

will

- Be
  - The:
    - Same.
- But
  - If
    - There:
      - Is

no

- Cycle
  - In:

$$G_d,$$

then

- There

    - Will
        - Be:
            - At least

one

- Vertex
    - That
        - Will:
            - Not

have

- An
    - Out
        - Going:
            - Edge.
- And
    - So
        - Only:

one

- Trim
    - Automaton:
        - Could

be

- Generated
    - From:

$$G_d.$$

- And

- So
  - ▹ Let:

$$\mathscr{A}_1, \qquad \mathscr{A}_2, \qquad \ldots \tag{5}$$

be

- The
  - Elements
    - ▹ Of:

$$C',$$

such that,

- The
  - Underlying:
    - ▹ Graphs

of

- All
  - Of
    - ▹ Them
      - Is:

$$G_1.$$

- Also
  - From
    - ▹ This,
      - We see that:

$$Q(\mathscr{A}_1) \quad = \quad Q(\mathscr{A}_2) \quad = \quad \ldots \quad = \quad Q \quad \text{(say)}.$$

- Let:

$$\mathscr{A}_{G_1} \quad = \quad \mathscr{A}_1 \quad \uplus \quad \mathscr{A}_2 \quad \uplus \quad \ldots$$

- Assume
  - That,
    - There:
      - Is

a

- Path from
  - The
    - Terminal
      - State of:

$$\mathscr{A}_1$$

to

- The
  - Initial:
    - State.
- Then
  - In
    - The:
      - List 5,

we see that,

- All
  - Elements
    - Of:

$$Q$$

will

- Be
    - An
        - Initial:
            - State

in

- At
    - Least
        - One:
            - Automaton,
- And
    - Also
        - A terminal
            - State

in

- At
    - Least
        - One:
            - Automaton

in

- The
    - Same:
        - List 5.
- And so
    - All

    - States
      - Of:

$$\mathscr{A}_{G_1}$$

will

- Be
  - Both:

initial *and* *terminal.*

- And
  - So:

$$\mathscr{A}_{G_1}$$

will

- Not
  - Be:
    - Minimal.
- And so
  - We
    - Remove
      - Some:

initial-states *and* *terminal-states*

- From:

$$\mathscr{A}_{G_1},$$

- Using:

δ *and* ζ,

so that,

- There
    - Will
        - Be:
            - Only

one

- State
    - In
        - Both:

$$I(\mathscr{A}_{G_1}) \qquad \textit{and} \qquad T(\mathscr{A}_{G_1}).$$

- Also
    - Since:

$$\delta$$

chooses

- An
    - Element
        - From:

$$I(\mathscr{A}_{G_1})$$

using

- The
    - Criterion:

“*minimum-value*,”

- And:

$$\zeta \qquad \textit{from}: \qquad T(\mathscr{A}_{G_1})$$

using

- The
  - ○ Criterion:

“*maximum-value*,”

we see that,

- The
  - ○ Two
    - ▷ Sets:

$$I(\mathscr{A}_{G_1}) \qquad \textit{and} \qquad T(\mathscr{A}_{G_1})$$

will

- Be
  - ○ Disjoint
    - ▷ After
      - Applying:

$$\delta \qquad \textit{and} \qquad \zeta.$$

- And
  - ○ So
    - ▷ Let:

$$\mathscr{A}_r \quad = \quad (Q(\mathscr{A}_{G_1}), \;\; I_r, \;\; T_r, \;\; Z(\mathscr{A}_{G_1}))$$

- Where:

$$I_r \quad = \quad \{ \;\; q_{i2} \;\; = \;\; \delta(\mathscr{A}_{G_1}, \;\; q_i) \;\; : \quad \textit{for all} \quad q_i \;\; \in \;\; I(\mathscr{A}_{G_1}) \;\; \},$$

$$T_r \quad = \quad \{ \;\; q_{t2} \;\; = \;\; \zeta(\mathscr{A}_{G_1}, \;\; q_t) \;\; : \quad \textit{for all} \quad q_t \;\; \in \;\; T(\mathscr{A}_{G_1}) \;\; \}.$$

- Also
    - Since
        - There:
            - Is

a

- Cycle
    - In:

$$\mathscr{A}_1,$$

we see that,

- All
    - States
        - In:

$$\mathscr{A}_r$$

can

- Be
    - Reached:
        - From

the

- Only
    - Initial:
        - State.
- And
    - So
        - There:

      - Will

be

- A path
  - ○ From
    - ▷ The initial:
      - State.

to

- All
  - ○ The
    - ▷ Other
      - States in:

$$\mathscr{A}_r.$$

- And
  - ○ Also
    - ▷ There:
      - Will

be

- A path
  - ○ From
    - ▷ All:
      - States

to

- The
  - ○ Only
    - ▷ Terminal:

            - State in:

$$\mathscr{A}_r.$$

- And
    - So:

$$\mathscr{A}_r$$

will

- Be
    - Trim.
        - ▷ And:
            - Also

since

- All
    - Edges
        - ▷ In:

$$\mathscr{A}_r$$

have

- Different
    - Labels,
        - ▷ From lemma 2,
            - We see that:

$$\mathscr{A}_r$$

will

- Be

    - Minimal
        - Regardless:
            - Of

the

- Number
    - Of
        - Elements:
            - In list 5.
- Let:

$$\mathscr{A}'$$

be

- The
    - Result
        - Of:
            - Combing

all

- Automata
    - In:

$$C'$$

- Using:

$$\uplus.$$

- Then from
    - The above
        - Argument,

      - We see that:

$$\mathscr{A}'$$

will

- Be
  - Trim.
    - Then:
      - If

we

- Apply:

$$\eta \qquad \textit{and} \qquad \theta$$

to

- Remove
  - Some
    - Elements
      - From:

$$\delta(I(\mathscr{A}')) \qquad \textit{and} \qquad \zeta(T(\mathscr{A}'))$$

to

- Get:

$$\mathscr{A},$$

which

- Is
  - The
    - Next
      - Step in:

κ

after

- Applying:

δ *and* ζ,

then

- No
    - Initial
        - ▹ State:
            - Can

be

- Reached
    - From
        - ▹ Another:
            - Initial-state,
- And
    - No
        - ▹ Terminal:
            - State

can

- Be reached
    - From
        - ▹ Another:
            - Terminal-state,
- And
    - Also:

$$I(\mathscr{A}) \qquad \textit{and} \qquad T(\mathscr{A})$$

will

- Be
    - ○ Disjoint.
        - ▹ And:
            - So

by

- That,
    - ○ The proof
        - ▹ Of
            - Minimality of:

$$\mathscr{A}$$

become

- Trivial.
    - ○ And
        - ▹ So:

$$\Theta(C)$$

will

- Be minimal.
    - ○ Note that,
        - ▹ The last
            - Step in:

$$\Theta$$

- Uses:

$$\kappa.$$

Lemma 4. *Let:*

$$\Theta(C) \quad = \quad \mathscr{A}.$$

- *Then:*

$$\mathscr{A}$$

- *Is:*

“minimal.”

- Let:

$$C_a \quad = \quad C_b, \qquad \alpha(C_a) \quad = \quad C'_a \qquad \text{and} \qquad \alpha(C_b) \quad = \quad C'_b.$$

- Then
  - From
    - Lemma 3,
      - We see that:

$$C'_a \quad = \quad C'_b.$$

- Let:

$$G_1$$

be

- A maximal
  - Biconnected
    - Subgraph
      - Of:

$$G_a.$$

- And

  - Let
    - The
      - Elements of:

$$C'_a$$

with

- The
  - Underlying
    - Graph:

$$G_1,$$

- Be:

$$\mathscr{A}_{a_1}, \qquad \mathscr{A}_{a_2}, \qquad \dots \tag{6}$$

- Then
  - Since:

$$C'_a \quad = \quad C'_b,$$

we see that,

- There
  - Exists:

$$\mathscr{A}_{b_1}, \qquad \mathscr{A}_{b_2}, \qquad \dots \tag{7}$$

- In:

$$C'_b,$$

- Such that:

$$|\mathscr{A}_{a_1}| \quad = \quad |\mathscr{A}_{b_1}|, \qquad |\mathscr{A}_{a_2}| \quad = \quad |\mathscr{A}_{b_2}|, \qquad \dots . \tag{8}$$

- Let:

$$\mathscr{A}_c \quad = \quad \mathscr{A}_{a_1} \quad \uplus \quad \mathscr{A}_{a_2} \quad \uplus \quad \ldots$$

and

$$\mathscr{A}_d \quad = \quad \mathscr{A}_{b_1} \quad \uplus \quad \mathscr{A}_{b_2} \quad \uplus \quad \ldots.$$

- Then
  - Since:

$$|\mathscr{A}_{a_1}| \quad = \quad |\mathscr{A}_{b_1}|$$

- We see that:

"*the Gödel-numbers*"

- Of:

initial-states *and* *terminal-states*

- In:

$\mathscr{A}_{a_1}$ *and* $\mathscr{A}_{b_1}$

constructed

- Using:

β *and* γ

will

- Be
  - The:
    - Same.
- And
  - So:

"*the Gödel-numbers*"

- Of:

initial-states *and* *terminal-states*

of

- All
    - Corresponding:
        - ▷ Automata

in

- The
    - Lists 6 and 7:
        - ▷ Will

also

- Be
    - The same.
        - ▷ And
            - So:

“*the Gödel-numbers*”

of

- The
    - Corresponding:

initial-states *and* *terminal-states*

- Of:

$\mathscr{A}_c$ *and* $\mathscr{A}_d$

will

- Also
  - Be
    - ▹ The:
      - Same.
- And
  - So
    - ▹ Use:

$$\delta \qquad \text{and} \qquad \zeta$$

to

- Remove
  - Some
    - ▹ States
      - From:

$$I(\mathscr{A}_c) \qquad \text{and} \qquad T(\mathscr{A}_c),$$

- And let
  - The
    - ▹ Resulting
      - Automaton be:

$$\mathscr{A}_e.$$

- And
  - Similarly,
    - ▹ Let:

$$\mathscr{A}_f$$

be

- The
    - Resulting:
        - Automaton

after

- Removing
    - Some
        - States
            - From:

$$I(\mathscr{A}_d) \qquad \textit{and} \qquad T(\mathscr{A}_d).$$

- Then
    - Since:

“*the Gödel-numbers*”

- Of:

initial-states *and* *terminal-states*

- Of:

$$\mathscr{A}_c \qquad \textit{and} \qquad \mathscr{A}_d$$

are

- The same,
    - And from
        - Statement 8,
            - We see that:

$$|\mathscr{A}_e| \quad = \quad |\mathscr{A}_f|.$$

- And

  - So
    - ▷ Use:

$$\eta \qquad \textit{and} \qquad \theta$$

to

- Remove
  - Some
    - ▷ States
      - From:

$$I(\mathscr{A}_e) \qquad \textit{and} \qquad T(\mathscr{A}_e),$$

- And
  - Let
    - ▷ The resulting
      - Automaton be:

$$\mathscr{A}_g.$$

- And
  - Similarly,
    - ▷ Construct:

$$\mathscr{A}_h$$

- From:

$$\mathscr{A}_f.$$

- Then
  - Since
    - ▷ There:
      - Is

a

- Mapping
  - Between:

“*the boundary-conditions*”

among

- All
  - Automata
    - ▷ In:

C_a *and* C_b,

we see that,

- For:

$$q_e \in I(\mathscr{A}_e),$$

if

- There
  - Is
    - ▷ No:
      - Path

from

- Any
  - Other
    - ▷ Element
      - Of:

$I(\mathscr{A}_e)$ *to*: q_e,

then

- There
    - Will
        - Be
            - A corresponding:

$$q_f \quad \in \quad I(\mathscr{A}_f),$$

such that,

- There
    - Will
        - Be:
            - No path

from

- Any
    - Other
        - Element
            - Of:

$$I(\mathscr{A}_f) \qquad \textit{to}: \qquad q_f,$$

and

vice-versa.

- And
    - So
        - From:
            - This,
- And

- Since:

$$|\mathscr{A}_e| = |\mathscr{A}_f|,$$

- We see that:

$$|\mathscr{A}_g| = |\mathscr{A}_h|.$$

- And
  - So
    - From:
      - This,
- And
  - From:
    - Lemmas 4 and 1,

we see that,

- The
  - Underlying
    - Graphs
      - Of:

$$\Theta(C_a) \qquad \textit{and} \qquad \Theta(C_b)$$

will

- Be:
  - Isomorphic.

Lemma 5. *Let:*

$$\begin{aligned} C_a &= C_b, \\ \Theta(C_a) &= \mathscr{A}_a, \\ \Theta(C_b) &= \mathscr{A}_b. \end{aligned}$$

- *Then:*

$$\widehat{\mathscr{A}_a} \cong \widehat{\mathscr{A}_b}.$$

- Let:

$$\Theta(C) = \mathscr{A}.$$

- Then
    - We see that,
        - ▷ A state,
            - Say:

$$q_i$$

will

- Be
    - An
        - ▷ Element
            - Of:

$$Q(\mathscr{A})$$

if

- And
    - Only:

- ▹ If

there

- Is
  - A vertex,
    - ▹ Say:

$$v_i \quad \in \quad V(G).$$

- And so
  - For
    - ▹ Every
      - State in:

$$Q(\mathscr{A}),$$

there

- Will be
  - A corresponding
    - ▹ Vertex
      - In:

$$V(G).$$

- And
  - So
    - ▹ There:
      - Will

be

- A mapping
  - Between

        - ▷ All states
            - In:

$$\mathscr{A}$$

- And
    - All
        - ▷ Vertices
            - In:

$$G.$$

- Let:

$$q_i \qquad \textit{and} \qquad q_j$$

be

- Two
    - States
        - ▷ In:

$$\mathscr{A}.$$

- Then
    - We see that,
        - ▷ There:
            - Will

be

- An
    - Edge
        - ▷ Between:

$$q_i \qquad \textit{and} \qquad q_j$$

if

- And
    - Only:
        - If

there

- Is
    - An
        - Edge
            - Between:

$$v_i \qquad \text{and} \qquad v_j.$$

- In:

$$G.$$

- And
    - So:
        - For

all

- Edges
    - In:

$$\mathscr{A}$$

there

- Will be
    - A corresponding
        - Edge
            - In:

$$G.$$

- And
    - So
        - There:
            - Will

be

- A mapping
    - Between
        - All
            - Edges in:

$$\mathscr{A} \qquad \text{and} \qquad G.$$

- And
    - So:

$$\widehat{\mathscr{A}} \;\cong\; G.$$

Lemma 6. *Let:*

$$\Theta(C) \;=\; \mathscr{A}.$$

- *Then:*

$$\widehat{\mathscr{A}} \;\cong\; G.$$

- Also
    - From:
        - Lemmas 5 and 6:

Theorem 2.1. *Let:*

$$C_a \quad = \quad C_b.$$

- *Then:*

$$G_a \quad \cong \quad G_b.$$

- Let:

$$G_a \quad \cong \quad G_b.$$

- And
  - Let:

$$G'_a \quad = \quad \Upsilon(G_a,\ m_1).$$

- Then
  - Since:

$$G_a \quad \cong \quad G_b$$

we see that,

- We can
  - Constructed
    - ▷ A Σ-GRAPH,
      - Say:

$$G'_a$$

with

- The
  - Underlying
    - ▷ Graph:

$$G_a,$$

- And
    - Another
        - Σ-GRAPH,
            - Say:

$$G'_b$$

with

- The
    - Underlying
        - Graph:

$$G_b,$$

- Such that:

$$G'_a \qquad \textit{and} \qquad G'_b$$

will

- Be
    - Isomorphic,
        - Even:
            - When

the

- Directions
    - And:
        - Labeling

are

- Also
    - Taken

        - ▷ Into:
            - Consideration.

- And
    - ○ So
        - ▷ Let:

$$C_a$$

be

- Generated
    - ○ From:

$$G'_a,$$

- And:

$$C_b$$

- From:

$$G'_b.$$

- Then
    - ○ Obviously,
        - ▷ We see that:

$$C_a \in G^\star_a, \qquad C_b \in G^\star_b \qquad \textit{and} \qquad C_a = C_b.$$

- Then
    - ○ We see that,
        - ▷ All:

$$C_a \in G^\star_a$$

is

- Constructed
  - From
    - Some:

$$G'_a \quad = \quad \Upsilon(G_a,\ m)$$

for

- Some
  - Value
    - Of:

$$m.$$

- And
  - So
    - For
      - All:

$$C_a \quad \in \quad G^\star_a$$

there

- Will
  - Exist:

$$C_b \quad \in \quad G^\star_b,$$

- Such that:

$$C_a \quad = \quad C_b.$$

Theorem 2.2. *Let:*

$$G_a \quad \cong \quad G_b.$$

- *Then*
  - *For*
    - *All:*

$$C_a \quad \in \quad G_a^\star,$$

there

- *Will*
  - *Exists:*

$$C_b \quad \in \quad G_b^\star,$$

- *Such that:*

$$C_a \quad = \quad C_b.$$

3 Ulam's conjecture

3.1 Sketch of proof

Let:

$$G \qquad \textit{and} \qquad H$$

be

- Two
  - ○ Isomorphic:
    - ▷ Graphs.
- And
  - ○ Let:

$$C_G \in G^\star, \qquad \textit{and} \qquad C_H \in H^\star.$$

- We
  - ○ Had
    - ▷ Saw
      - That:

$$C_G = C_H \qquad \textit{implies}: \qquad G \cong H.$$

- And
  - ○ So
    - ▷ Let:

“*the Ulam's-deck*”

- Of:

$$G \qquad \textit{and} \qquad H$$

- Be:

$$\{\ G_1,\ G_2,\ \dots,\ G_u\ \}$$

and

$$\{\ H_1,\ H_2,\ \dots,\ H_u\ \}.$$

- Then
    - ○ From
        - ▹ The new:
            - Algorithm,

the

- Conjecture
    - ○ Can
        - ▹ Be
            - Stated as:

$$C_{G_1} = C_{H_1}, \qquad C_{G_2} = C_{H_2}, \qquad \dots, \qquad C_{G_u} = C_{H_u}$$

implies

$$G \cong H.$$

- Let:

$$C'_G = C_{G_1} \cup C_{G_2} \cup \dots \cup C_{G_u}$$

and

$$C'_H = C_{H_1} \cup C_{H_2} \cup \dots \cup C_{H_u}.$$

- We
    - ○ Then
        - ▹ Show
            - That:

$$C_G \quad \textit{and} \quad C_H$$

can

- Be
    - ○ Constructed
        - ▷ From:

$$C'_G \quad \textit{and} \quad C'_H$$

respectively,

- Or
    - ○ First
        - ▷ We
            - – Apply:

$$\mu$$

- On:

$$C'_G \quad \textit{and} \quad C'_H,$$

until

- No
    - ○ More:
        - ▷ Changes

can

- Be
    - ○ Made.
        - ▷ And:
            - – Then

show

- That
    - The
        - ▷ Results:
            - Of

those

- Two
    - Iterative
        - ▷ Applications:
            - Are equal

if

- And
    - Only
        - ▷ If:

$$G \quad \cong \quad H.$$

3.2 Definitions

We

- Define:

$$D(G)$$

- As:

“*The Ulam’s-deck*”

- Of:

$$G.$$

- Formally:

$$D(G)$$

is

- Defined
    - To
        - Be:

$$\{ \quad G_i \quad : \quad G_i \quad = \quad G - v_i \quad \text{for all} \quad v_i \ \in \ V(G) \quad \}.$$

- For
    - Two
        - Graphs,
            - Say:

$$G \qquad \textit{and} \qquad H,$$

the

- Function:

$$M$$

is

- Used
    - To:
        - Define

the

- Mapping
    - Between:

$$D(G) \qquad \textit{and} \qquad D(H)$$

as

- Defined
    - In
        - ▷ The:
            - Conjecture.
- Formally:

$$M(D(G), \quad D(H))$$

is

- Defined
    - To
        - ▷ Be:
            - True,

if

- And
    - Only
        - ▷ If:

$$\textit{for all}: \;\; G_i \in D(G) \quad \textit{there exists}: \;\; H_j \in D(H) \quad \textit{such that}: \quad G_i \;\; \cong \;\; H_j$$

$$\textit{and}$$

$$\textit{for all}: \;\; H_j \in D(H) \quad \textit{there exists}: \;\; G_i \in D(G) \quad \textit{such that}: \quad H_j \;\; \cong \;\; G_i$$

is

- True.
    - Let:

$$G_i \;\; \in \;\; D(G).$$

- Then

- We see that,
  - Even:
    - Though,

the

- Number
  - Of
    - Vertices
      - In:

$$G_i$$

is

- One
  - Less
    - Than
      - That in:

$$G,$$

- All:

$$C_i \in G_i^\star$$

are

- Also
  - Constructed
    - Using:

$$\Sigma.$$

- Let:

$$\Omega \quad : \quad \mathbf{P}(S_G) \quad \times \quad \mathbf{N} \quad \to \quad \underbrace{\mathbf{P}(S_{\mathscr{A}}) \times \cdots \times \mathbf{P}(S_{\mathscr{A}})}_{u\ \textit{times}},$$

$$D(G) \quad = \quad \{\ G_1, \quad G_2, \quad \ldots, \quad G_u\ \}.$$

- And
  - Let:

$$C_1 \ \in \ G_1^\star, \qquad C_2 \ \in \ G_2^\star, \qquad \ldots \qquad C_u \ \in \ G_u^\star.$$

- Then:

$$\Omega(D(G), \quad m) \quad = \quad (C_1, \quad C_2, \quad \ldots, \quad C_u),$$

- If:

“*the Gödel-number*”

- Of:

$$(C_1, \quad C_2, \quad \ldots, \quad C_u)$$

is

- Equal
  - To:

$$m,$$

- And
  - Undefined
    - ▷ For all:
      - Other cases.
- Let:

$$e_1 \ \in \ Z(\mathscr{A}_1) \qquad \textit{and} \qquad e_2 \ \in \ Z(\mathscr{A}_2).$$

- Then:

$$e_1 \qquad \textit{and} \qquad e_2$$

are

- Said
  - To
    - ▹ Be:

SAME-STATE-NAME-EDGES,

if

- And
  - Only
    - ▹ If:

$$\pi_1(e_1) = \pi_1(e_2) \qquad \textit{and} \qquad \pi_3(e_1) = \pi_3(e_2)$$

or

$$\pi_1(e_1) = \pi_3(e_2) \qquad \textit{and} \qquad \pi_3(e_1) = \pi_1(e_2).$$

- Exemplifying,
  - If:

$$e_1$$

- Is:

$$(q_1, \quad \sigma, \quad q_2)$$

- And:

$$e_2$$

- Is:

$$(q_1, \quad \sigma, \quad q_2),$$

- Then:

$$e_1 \qquad\qquad \textit{and} \qquad\qquad e_2$$

- Are:

SAME-STATE-NAME-EDGES.

- Also
  - If:

$$e_1$$

- Is:

$$(q_1, \quad \sigma, \quad q_2)$$

- And:

$$e_2$$

- Is:

$$(q_2, \quad \sigma_1, \quad q_1),$$

- Then:

$$e_1 \qquad\qquad \textit{and} \qquad\qquad e_2$$

- Are:

SAME-STATE-NAME-EDGES.

- But
  - If:

$$e_1$$

- Is:

$$(q_1, \quad \sigma, \quad q_2)$$

- And:

$$e_2$$

- Is:

$$(q_1, \quad \sigma_1, \quad q_3),$$

- Then:

$$e_1 \qquad \textit{and} \qquad e_2$$

are

- Not:

SAME-STATE-NAME-EDGES.

- Let:

$$D(G) \quad = \quad \{ \; G_1, \quad G_2, \quad \ldots, \quad G_u \; \}.$$

- And
  - Let:

$$C_1 \;\in\; G_1^\star, \qquad C_2 \;\in\; G_2^\star, \qquad \ldots \qquad C_u \;\in\; G_u^\star.$$

- And
  - Let:

$$C' \quad = \quad C_1 \;\cup\; C_2 \;\cup\; \ldots \;\cup\; C_u.$$

- We
  - Had
    - Earlier:
      - Stated

that,

- We
    - Will
        - Be:
            - Constructing

an

- Element
    - Of:

$$G^\star$$

- From:

$$C'.$$

- And
    - So
        - Let:

$$\mathscr{A}_1, \quad \mathscr{A}_2 \quad \in \quad C'.$$

- And
    - Let:

$$e_1 \quad \in \quad Z(\mathscr{A}_1) \qquad \textit{and} \qquad e_2 \quad \in \quad Z(\mathscr{A}_2).$$

- Also
    - Assume
        - That:

$$e_1 \qquad \textit{and} \qquad e_2$$

- Are:

SAME-STATE-NAME-EDGES.

- Then
  - ○ We see that,
    - ▹ If:

$$e_1 \qquad\qquad \textit{and} \qquad\qquad e_2$$

have

- Different
  - ○ Representations,
    - ▹ Then:
      - It

will

- Be
  - ○ Obvious
    - ▹ That:
      - We

cannot

- Construct
  - ○ An
    - ▹ Element
      - Of:

$$G^\star$$

- From:

$$C'.$$

- And so

- We
  - ▷ Say
    - That:

$$\mathscr{A}_1 \quad \odot \quad \mathscr{A}_2,$$

if

- And
  - Only
    - ▷ If
      - For all:

$$e_1 \quad \in \quad Z(\mathscr{A}_1)$$

- And
  - For
    - ▷ All:

$$e_2 \quad \in \quad Z(\mathscr{A}_2),$$

- If:

$$e_1 \qquad \textit{and} \qquad e_2$$

- Are:

$$\textsc{Same-State-Name-Edges},$$

- Then:

$$e_1 \qquad \textit{and} \qquad e_2$$

will

- Have
  - The

    - ▹ Same:
        - – Representations.

- Let:

$$\Omega(D(G), \quad m) \quad = \quad CL \quad = \quad (C_1, \quad C_2, \quad \ldots, \quad C_u).$$

- And
    - ◦ Let:

$$C' \quad = \quad C_1 \ \cup \ C_2 \ \cup \ \ldots \ \cup \ C_u.$$

- We
    - ◦ Say
        - ▹ That:

$$CL \qquad \textit{satisfies}: \qquad \diamondsuit,$$

if

- And
    - ◦ Only
        - ▹ If:

$$\textit{for all}: \quad \mathscr{A}_1, \ \mathscr{A}_2 \ \in \ C', \quad \mathscr{A}_1 \ \odot \ \mathscr{A}_2,$$

- For
    - ◦ Two
        - ▹ Graphs,
            - – Say:

$$G \qquad \textit{and} \qquad H,$$

the

- Conjecture

    - States:
        - That,

if

- There
    - Is
        - An:
            - Isomorphism

between

- Each of
    - The
        - Graphs
            - In:

$$D(G) \qquad \textit{and} \qquad D(H),$$

- Then:

$$G \;\cong\; H.$$

- And
    - So:
        - We

can

- Assume
    - That
        - The:
            - Isomorphism

between

- All
    - Elements
        - Of:

$D(G)$ *and* $D(H)$

is

- Given.
    - And
        - So:

$D(G)$ *and* $D(H)$

is

- Given.
    - Then
        - Since
            - All:

vertices *and* *edges*

- Of:

G

- Is:

a vertex *or* *an edge*

of

- Some
    - Element
        - Of:

$$D(G),$$

we

- Assume
    - That,
        - When:

$$D(G) \qquad \textit{and} \qquad D(H),$$

are

- Given,
    - The
        - Graphs:

$$G \qquad \textit{and} \qquad H$$

have

- Been
    - Constructed
        - From:
            - Them

without

- Explicitly
    - Mentioning
        - Any:
            - Construction.
- Let:

$$D(G) \quad = \quad \{ \ G_1, \ \ G_2, \ \ \ldots, \ \ G_u \ \}.$$

- The

- Function:

$$\Phi$$

is

- Used
  - To
    - ▹ Assign:

a direction *and* *a label*

to

- All edges
  - Of
    - ▹ All graphs
      - In:

$$D(G),$$

- Such that:

“*an u-tuple*”

- Satisfying:

$$\diamondsuit$$

can

- Be
  - Constructed.
    - ▹ Let:

$$\Phi \quad : \quad \mathbf{P}(S_G) \quad \times \quad \mathbf{N} \quad \to \quad \mathbf{P}(S_\Sigma),$$

$$\Upsilon(G, \quad m_d) \quad = \quad G_d,$$

$$D(G) \quad = \quad \{ \; G_1, \;\; G_2, \;\; \ldots, \;\; G_u \;\; \},$$

- And
    - Let:

$$G_{d_1} \quad \subset \quad G_d,$$

- Such that:

$$G_1$$

is

- The
    - Underlying
        - ▷ Graph
            - Of:

$$G_{d_1},$$

- And
    - Let:

$$G_{d_2} \quad \subset \quad G_d,$$

- Such that:

$$G_2$$

is

- The
    - Underlying
        - ▷ Graph
            - Of:

$$G_{d_2},$$

$\vdots$

- And
  - Let:

$$G_{d_u} \quad \subset \quad G_d,$$

- Such that:

$$G_u$$

is

- The
  - Underlying
    - Graph
      - Of:

$$G_{d_u}.$$

- Then:

$$\Phi(D(G), \quad m) \quad = \quad \{ \; G_{d_1}, \quad G_{d_2}, \quad \ldots, \quad G_{d_u} \; \},$$

- If:

“*the Gödel-number*”

- Of:

$$\{ \; G_{d_1}, \quad G_{d_2}, \quad \ldots, \quad G_{d_u} \; \}$$

is

- Equal
  - To:

$$m,$$

- And
  - Undefined
    - ▹ For:
      - Other cases.
- Let:

$$\Phi(D(G), \quad m_d) \quad = \quad \{ \; G_{d_1}, \quad G_{d_2}, \quad \ldots, \quad G_{d_u} \; \}.$$

- Then
  - We see that,
    - ▹ For:
      - All pair

of

- Graphs,
  - Say:

$$G_{d_i}, \quad G_{d_j} \quad \in \quad \Phi(D(G), \quad m_d),$$

- And
  - For:

$$e_i \quad \in \quad E(G_{d_i}) \qquad \textit{and} \qquad e_j \quad \in \quad E(G_{d_j}),$$

- If:

$$e_i \qquad \textit{and} \qquad e_j$$

- Are:

“SAME-STATE-NAME-EDGES”

- Then:

the directions *and* *labels*

in

- Both
    - ○ Of:
        - ▷ Them

will

- Also
    - ○ Be
        - ▷ The:
            - Same.
- And
    - ○ So
        - ▷ If:

$$C_{d_1} \quad = \quad \Gamma(G_{d_1}), \qquad C_{d_2} \quad = \quad \Gamma(G_{d_2}), \qquad \dots \qquad C_{d_u} \quad = \quad \Gamma(G_{d_u}),$$

then

- It
    - ○ Is
        - ▷ Obvious
            - That:

$$(C_{d_1}, \quad C_{d_2}, \quad \dots, \quad C_{d_u}) \qquad \textit{satisfies}: \qquad \diamondsuit.$$

Lemma 7. *There*

- *Exists:*

$$\text{“}a \quad \Omega(D(G), \quad m)\text{,”}$$

such that,

- *It*

  - *Satisfies:*

$$\diamondsuit.$$

- Let:

$$\Omega(D(G), \quad m) \quad = \quad CL \quad = \quad (C_1, \quad C_2, \quad \ldots, \quad C_u).$$

- Then
  - From:
    - ▹ Lemma 7,

assume

- That:

$$CL \qquad \textit{satisfies}: \qquad \diamondsuit,$$

- And
  - Let:

$$C' \quad = \quad C_1 \quad \cup \quad C_2 \quad \cup \quad \ldots \quad \cup \quad C_u.$$

- Then
  - Since:

$$CL \qquad \textit{satisfies}: \qquad \diamondsuit,$$

we see that,

- There
  - Will
    - ▹ Exist:

$$C \quad \in \quad G^{\star},$$

- Such that:

$$C' \quad \subseteq \quad C.$$

- But
    - In
        - Some:
            - Cases,

$$C'$$

will

- Be
    - A proper
        - Subset
            - Of:

$$C,$$

since

- There
    - Maybe
        - automaton
            - In:

$$C,$$

generated

- From
    - The
        - Whole
            - Graph:

$$G,$$

- And
    - That:
        - ▷ Automaton

will

- Not be
    - An
        - ▷ Element
            - Of:

$$C',$$

- Exemplifying,
    - Let:

$$\mathscr{A}_1 \in C_1 \qquad \textit{and} \qquad \mathscr{A}_2 \in C_2,$$

- And
    - Let:

$$T(\mathscr{A}_1) = I(\mathscr{A}_2),$$

- And
    - Let:

$$\mathscr{A} = (Q(\mathscr{A}_1) \cup Q(\mathscr{A}_2), \quad I(\mathscr{A}_1), \quad T(\mathscr{A}_2), \quad Z(\mathscr{A}_1) \cup Z(\mathscr{A}_2)).$$

- And
    - Let:

$$G$$

be

- The

    - Underlying
        - Graph
            - Of:

$$\mathscr{A}.$$

- Then
    - We see that:

$$\mathscr{A}$$

will

- Not be
    - An
        - Element
            - Of:

$$C',$$

- But
    - It
        - Will:
            - Certainly

be

- An
    - Element
        - Of:

$$C,$$

- And
    - So:

$$\succ, \qquad \|, \qquad \circlearrowright, \qquad \oplus \qquad \textit{and} \qquad \otimes$$

are

- Used to
    - Construct
        - New automat
            - From:

$$C',$$

so that,

- It
    - Will
        - Be:
            - Possible

to

- Construct
    - An
        - Element
            - Of:

$$G^{\star}$$

- From:

$$C'.$$

- Let:

$$`\succ' \qquad `\|' \qquad \textit{and} \qquad `\circlearrowright'$$

be

- Subsets

    - Of:

$$S_{\mathscr{A}} \quad \times \quad S_{\mathscr{A}}.$$

- Then:

$$\mathscr{A}_1 \quad \succ \quad \mathscr{A}_2$$

will

- Hold
    - If
        - And
            - Only if:

$$\begin{aligned} T(\mathscr{A}_1) &\quad = \quad I(\mathscr{A}_2), \\ I(\mathscr{A}_1) &\quad \neq \quad T(\mathscr{A}_2), \end{aligned}$$

- And:

$$\mathscr{A}_1 \quad \| \quad \mathscr{A}_2$$

will

- Hold
    - If
        - And
            - Only if:

$$\begin{aligned} I(\mathscr{A}_1) &\quad = \quad I(\mathscr{A}_2), \\ T(\mathscr{A}_1) &\quad = \quad T(\mathscr{A}_2), \end{aligned}$$

- And:

$$\mathscr{A}_1 \quad \circlearrowright \quad \mathscr{A}_2$$

will

- Hold
  - If
    - ▹ And
      - Only if:

$$\begin{aligned} T(\mathscr{A}_1) &= I(\mathscr{A}_2), \\ I(\mathscr{A}_1) &= T(\mathscr{A}_2), \end{aligned}$$

- And:

$$\mathscr{A}_1 \oplus \mathscr{A}_2$$

is

- Defined
  - To
    - ▹ Be:

$$\mathscr{A} \quad = \quad (Q, \quad I(\mathscr{A}_1), \quad T(\mathscr{A}_2), \quad Z),$$

- Where:

$$Q \quad = \quad Q(\mathscr{A}_1) \quad \cup \quad Q(\mathscr{A}_2),$$

$$Z \quad = \quad Z(\mathscr{A}_1) \quad \cup \quad Z(\mathscr{A}_2),$$

- And:

$$\mathscr{A}_1 \otimes \mathscr{A}_2$$

is

- Defined
  - To
    - ▹ Be:

$$\mathscr{A} \quad = \quad (Q, \quad I(\mathscr{A}_1), \quad T(\mathscr{A}_1), \quad Z),$$

- Where:

$$Q \quad = \quad Q(\mathscr{A}_1) \quad \cup \quad Q(\mathscr{A}_2),$$

$$Z \quad = \quad Z(\mathscr{A}_1) \quad \cup \quad Z(\mathscr{A}_2).$$

- Let:

$$\Omega(D(G), \quad m) \quad = \quad (C_1, \quad C_2, \quad \ldots, \quad C_u)$$

be

- Such that,
  - It
    - Satisfies:

$$\diamondsuit.$$

- And
  - Also
    - Let:

$$C' \quad = \quad C_1 \quad \cup \quad C_2 \quad \cup \quad \ldots \quad \cup \quad C_u.$$

- The
  - Function:

$$\mu$$

is

- Used
  - To:
    - Add

new

- Automata
  - Constructed
    - Using:

$$\oplus, \qquad \uplus \qquad \textit{and} \qquad \otimes$$

- Into:

$$C'.$$

- Formally,
  - Let:

$$\mu \quad : \quad \mathbf{P}(S_{\mathscr{A}}) \quad \to \quad \mathbf{P}(S_{\mathscr{A}}),$$

$$S_a \quad \in \quad \mathbf{P}(S_{\mathscr{A}}),$$

- And:

$$S_b \quad = \quad \{ \quad \mathscr{A} \quad = \quad \mathscr{A}_1 \oplus \mathscr{A}_2 \quad : \quad \mathscr{A}_1 \quad \succ \quad \mathscr{A}_2 \quad \textit{for all} \quad \mathscr{A}_1, \; \mathscr{A}_2 \quad \in \quad S_a \, \},$$

$$S_b \quad = \quad \{ \quad \mathscr{A} \quad = \quad \mathscr{A}_1 \uplus \mathscr{A}_2 \quad : \quad \mathscr{A}_1 \quad \| \quad \mathscr{A}_2 \quad \textit{for all} \quad \mathscr{A}_1, \; \mathscr{A}_2 \quad \in \quad S_a \, \},$$

$$S_b \quad = \quad \{ \quad \mathscr{A} \quad = \quad \mathscr{A}_1 \otimes \mathscr{A}_2 \quad : \quad \mathscr{A}_1 \quad \circlearrowright \quad \mathscr{A}_2 \quad \textit{for all} \quad \mathscr{A}_1, \; \mathscr{A}_2 \quad \in \quad S_a \, \}.$$

- Then:

$$\mu(S_a) \quad = \quad S_a \quad \cup \quad S_b \quad \cup \quad S_c \quad \cup \quad S_d.$$

- Note that,
  - The
    - Value
      - Returned by:

$$\mu$$

may

- Not
  - Be
    - An
      - Element of:

$$G^{\star}.$$

- And
  - So
    - In:

$$\Psi,$$

we

- Iterate
  - Using:

$$\mu$$

so that,

- We will
  - Get
    - An
      - Element of:

$$G^{\star}.$$

- Formally,
  - Let:

$$\Psi \quad : \quad \mathbf{P}(S_{\mathscr{A}}) \quad \rightarrow \quad \mathbf{P}(S_{\mathscr{A}}),$$

$$S_a \quad \in \quad \mathbf{P}(S_{\mathscr{A}}).$$

- And
    - Let:

$$\begin{aligned} S_1 &= S_a, \\ S_2 &= \mu(S_1), \\ &\vdots \\ S_i &= \mu(S_{i-1}), \\ &\vdots \end{aligned}$$

- Then:

$$\Psi(C) = \bigcup_{i\,=\,1}^{\infty} S_i.$$

- Note that:

$$\Psi$$

uses

- Only:

$$\mu \tag{9}$$

while

- Iterating,
    - And
        - The:
            - Function 9

uses

- Only:

$$\oplus, \qquad \uplus \qquad \textit{and} \qquad \otimes\,. \tag{10}$$

- And
    - Operators
        - In:
            - the list 10

does

- Not
    - Produce
        - Any
            - New:

$$\textit{state} \qquad \textit{or} \qquad \textit{edge}$$

that

- Is
    - Not:
        - Present

in

- An
    - Automata
        - Of:

$$S_{\mathscr{A}}\,.$$

- And
    - So:
        - After

a

- Finite
    - Number
        - Of
            - Iterations:

S_i *in* Ψ

will

- Not
    - Change.
        - And:
            - So

the

- Value
    - Returned
        - By:

Ψ

will

- Be
    - Finite.
        - And:
            - Also

since

- No
    - New:

state *or* *edge*

is

- Generated,
  - We see that,
    - The value
      - Returned by:

$$\Psi$$

will

- Be
  - A subset
    - Of:

$$S_{\mathscr{A}}.$$

Lemma 8. *Let:*

$$C \quad \subset \quad S_{\mathscr{A}}.$$

- *Then:*

$$\Psi(C) \quad \subset \quad S_{\mathscr{A}}.$$

3.3 Proving the conjecture

Let:

$$\Omega(D(G), \quad m) \quad = \quad CL \quad = \quad (C_1, \quad C_2, \quad \ldots, \quad C_u).$$

- Then
  - From:
    - Lemma 7,

assume

- That:

$$CL \qquad \textit{satisfies}: \qquad \diamondsuit.$$

- And
    - So
        - ▹ From:
            - This,

we see that,

- There
    - Will:
        - ▹ Exist

“*a* Σ-GRAPH,”

- Say:

$$G_d,$$

- Such that:

$$G_{d_1} \subset G_d, \qquad G_{d_2} \subset G_d, \qquad \ldots, \qquad G_{d_u} \subset G_d, \quad (11)$$

and

$$C_1 = \Gamma(G_{d_1}), \qquad C_2 = \Gamma(G_{d_2}), \qquad \ldots, \qquad C_u = \Gamma(G_{d_u}). \quad (12)$$

- And
    - So
        - ▹ Assume
            - That:

$$C = \Gamma(G_d).$$

- And
    - Let:

$$C' \quad = \quad C_1 \quad \cup \quad C_2 \quad \cup \quad \ldots \quad \cup \quad C_u.$$

- Then
    - From
        - Equations 11 and 12,
            - We see that:

$$C' \quad \subset \quad C_d.$$

- Then
    - We see that,
        - It:
            - Maybe

that

- For:

$$\mathscr{A}_1, \quad \mathscr{A}_2 \quad \in \quad C',$$

the

- Relation:

$$\mathscr{A}_1 \quad \succ \quad \mathscr{A}_2$$

may

- Hold.
    - And
        - So:
            - If

we

- Perform:

$$\Psi(C'),$$

- Then:

$$\mathscr{A}_1 \quad \oplus \quad \mathscr{A}_2$$

will

- Be
    - ○ Generated.
        - ▹ From:
            - – Equation 11,

we see that,

- The
    - ○ Underlying
        - ▹ Graphs
            - – Of:

$$\mathscr{A}_1 \quad \oplus \quad \mathscr{A}_2$$

will

- Be
    - ○ A subgraph
        - ▹ Of:

$$G_d.$$

- And
    - ○ So
        - ▹ We see that:

$$\mu(C') \quad \subset \quad C_d.$$

- And
    - ○ So

    - Let:

$$C'' \quad = \quad C' \;\cup\; \mu(C').$$

- Then
  - We see that,
    - It:
      - Maybe

that,

- For:

$$\mathscr{A}_1, \;\; \mathscr{A}_2 \;\; \in \;\; C'',$$

the

- Relations:

$$\mathscr{A}_1 \;\succ\; \mathscr{A}_2, \qquad \mathscr{A}_1 \;\|\; \mathscr{A}_2, \qquad \textit{and} \qquad \mathscr{A}_1 \;\circlearrowright\; \mathscr{A}_2$$

may

- Hold.
  - And
    - So:
      - If

we

- Perform:

$$\mu(C''),$$

then

- Some
  - New:

      - ▷ Automata

will

- Be
  - ○ Generated.
    - ▷ And:
      - So

using

- The
  - ○ Same
    - ▷ Line:
      - Of arguments

which

- We
  - ○ Gave
    - ▷ Earlier,
      - We see that:

$$\mu(C'') \quad \subset \quad C_d.$$

- Then
  - ○ Since
    - ▷ The:
      - Number

of

- Different
  - ○ State
    - ▷ Names:

            - Is finite,

we see that,

- There
    - Will be
        - A finite:
            - Limit

to

- Number
    - Of:
        - Relations

that

- Can
    - Hold.
        - And:
            - So

after

- A finite
    - Number
        - Of:
            - Iterations

we see that,

- No
    - New
        - Automata:
            - Will

be

- Added
    - Into:

$$C^k \quad = \quad \mu(C^{k-1}),$$

- And
    - Also:

$$C^k \quad \subset \quad C_d.$$

- But if
    - No
        - ▹ New:
            - Automata

can

- Be
    - Generated
        - ▹ By:
            - Considering

the

- Relations
    - Between
        - ▹ The
            - Names of:

initial-state *and* *terminal-state*

of

- All:

$$\mathscr{A} \quad \in \quad C^k,$$

then

- It
    - Would:
        - Mean

that,

- All
    - Possible
        - Combinations
            - Using:

$$(\succ, \quad \oplus), \qquad (\|, \quad \uplus), \qquad \textit{and} \qquad (\circlearrowright, \quad \otimes),$$

- And
    - All
        - Edges
            - Of all:

$$\mathscr{A} \quad \in \quad C^k,$$

would

- Have
    - Been:
        - Generated,
- And
    - Added
        - Into:

$$C^k.$$

- Also
    - Since
        - For edges
            - In:

$$E(G_d),$$

there

- Is
    - A corresponding
        - Automaton:
            - with

one

- Edge
    - In:

$$C',$$

- And
    - Since:

$$\succ, \qquad \|, \qquad \textit{and} \qquad \circlearrowright$$

can

- Define
    - All
        - Possible:
            - Name-relations

between

- All:

initial-states *and* *terminal-states*

in

- All
    - Automata
        - Of:

$$C',$$

we see that,

- The
    - Duos:

$$(\succ, \quad \oplus), \qquad (\|, \quad \uplus), \qquad (\circlearrowright, \quad \otimes),$$

can

- Be
    - Used
        - To:
            - Define

all

- Possible
    - Extensions:
        - To

all

- The
    - Already
        - Existing:
            - Paths,

- And
    - Also
        - ▷ All:
            - Combinations

of

- All
    - The already
        - ▷ Existing:
            - Paths

using

- The edges
    - In
        - ▷ All
            - Automata of:

$$C'.$$

- And
    - So:
        - ▷ When

no

- New automata
    - Can
        - ▷ Be added
            - Into:

$$C^k,$$

we see that,

- All
    - Combinations
        - Of:
            - All paths

between

- All
    - Pair
        - Of:
            - States

in

- All
    - Automata
        - Of:

$$C'$$

would

- Have
    - Been:
        - Generated,
- And
    - Added
        - Into:

$$C^k.$$

- And
    - So

      - ▹ From:
        - This,

- And
  - ○ Since
    - ▹ All:
      - Paths

corresponds

- To
  - ○ A behavior,
    - ▹ And

since

- There
  - ○ Is
    - ▹ An:
      - Automaton

for

- All
  - ○ Combinations
    - ▹ Of:
      - Behaviors,
- And
  - ○ Since
    - ▹ For:
      - All edges

in

- An
    - Automata
        - Of:

$$C^k,$$

there

- Is
    - A corresponding
        - Edge
            - In:

$$E(G_d),$$

- And
    - Since:

$$E(G_d) \quad = \quad E(G_{d_1}) \quad \cup \quad E(G_{d_2}) \quad \cup \quad \ldots \quad \cup \quad E(G_{d_u}),$$

- And
    - Similarly
        - For
            - All:

“*states,*”

we see that,

- All
    - Possible:
        - Automata

that

- Can

    - Be
        - ▹ Generated
            - From:

$$G_d,$$

would

- Have
    - Been:
        - ▹ Generated,
- And
    - Added
        - ▹ Into:

$$C^k.$$

- And
    - So
        - ▹ From:
            - This,
- And
    - Since:

$$C^k \quad \subset \quad C_d,$$

- We see that:

$$C^k \qquad = \qquad C_d.$$

- And
    - So:

$$\Psi(C') \quad = \quad C_d,$$

- And
    - So:

$$\Psi(C')$$

will

- Be
    - An
        - Element
            - Of:

$$G^{\star}.$$

Lemma 9. *Let:*

$$\Omega(D(G), \quad m) \qquad (C_1, \quad C_2, \quad \ldots, \quad C_u),$$

be

- *Such that,*
    - *It*
        - *Satisfies:*

$$\diamondsuit.$$

- *And*
    - *Let:*

$$C' \quad = \quad C_1 \ \cup \ C_2 \ \cup \ \ldots \ \cup \ C_u.$$

- *Then:*

$$\Psi(C') \ \in \ G^{\star}.$$

- Note that,

    - When:
        - A vertex

is

- Deleted
    - From:
        - A graph,

the

- Resulting
    - Graph
        - Maybe:
            - Disconnected.
- And
    - Also
        - One:
            - Component

of

- The
    - Disconnected:
        - Graph

may

- Have
    - Only
        - One:
            - Vertex,
- And

    - No
        - ▷ Edge:
            - In it.

- And
    - So
        - ▷ In:
            - Such cases,

we see that,

- No
    - Automaton:
        - ▷ Will

be

- Generated
    - From
        - ▷ That:
            - Component

with

- Just
    - One:
        - ▷ Vertex,
            - Since:

$$\Xi(\Upsilon(G,\ m),\ \ v_i,\ \ v_j,\ \ k)$$

is

- Defined
    - Only

            - When:

$$v_i \qquad \textit{and} \qquad v_j$$

are

- Distinct,
    - And
        - When:
            - There

is

- At least
    - One
        - Path
            - From:

$$v_i \qquad \textit{to}: \qquad v_j.$$

- Also
    - From
        - Now:
            - Onwards,

we

- Assume
    - That:

$$G \qquad \textit{and} \qquad H$$

are

- Graphs,
    - Such that,

        - ▷ The:
            - Number

of

- Vertices
    - ○ In
        - ▷ Both:
            - Of them

are

- Equal,
    - ○ And
        - ▷ Also
            - Greater than:

$$8.$$

- Let:

$$\begin{array}{rcl} D(G) & = & \{\ G_1,\ \ G_2,\ \ \ldots,\ \ G_u\ \}, \\ D(H) & = & \{\ H_1,\ \ H_2,\ \ \ldots,\ \ H_u\ \}. \end{array}$$

- Also
    - ○ Assume:
        - ▷ That,

$$M(D(G),\quad D(H))$$

is

- True.
    - ○ And
        - ▷ Let:

$$\Omega(D(G), \quad m_G) \quad = \quad CL_G \quad = \quad (C_{G_1}, \quad C_{G_2}, \quad \ldots, \quad C_{G_u}),$$

- Also
  - Assume
    - ▹ That:

$$CL_G \qquad \textit{satisfies}: \qquad \diamondsuit.$$

- Then
  - From:
    - ▹ Theorem 2.2,
- And
  - Since:

$$M(D(G), \quad D(H))$$

is

- True,
  - We see that,
    - ▹ There:
      - Exists

$$CL_H,$$

- Such that:

$$CL_H \quad = \quad (C_{H_1}, \quad C_{H_2}, \quad \ldots, \quad C_{H_u})$$

and

$$C_{G_1} \quad = \quad C_{H_1}, \qquad C_{G_2} \quad = \quad C_{H_2}, \qquad \ldots, \qquad C_{G_u} \quad = \quad C_{H_u}.$$

- Also
  - Since:

$$CL_G \quad \textit{satisfies} \quad \diamondsuit,$$

- We see that:

$$CL_H$$

will

- Also
    - Satisfy:

$$\diamondsuit.$$

- Then
    - Since
        - ▷ The:
            - Number

of

- Vertices
    - In
        - ▷ Both:

$$G \quad \textit{and} \quad H$$

are

- Greater
    - Than:

$$8,$$

we

- Can
    - Assume

        - ▷ That:

$$\text{“}C_{G_1} \quad \textit{and} \quad C_{G_2} \qquad\qquad \textit{intersects,”}$$

- And
    - So
        - ▷ Does:

$$C_{H_1} \qquad\qquad \textit{and} \qquad\qquad C_{H_2}.$$

- Also
    - Since:

$$C_{G_1} \;=\; C_{H_1} \qquad\qquad \textit{and} \qquad\qquad C_{G_2} \;=\; C_{H_2}$$

- We see that:

$$C_{G_1} \;\cap\; C_{G_2} \quad = \quad C_{H_1} \;\cap\; C_{H_2}.$$

- And
    - So
        - ▷ For
            - All:

$$\mathscr{A} \quad \in \quad C_{G_1} \;\cap\; C_{G_2},$$

there

- Will be
    - An
        - ▷ Automaton
            - In:

$$C_{H_1} \;\cap\; C_{H_2}$$

with

- The
    - Same:
        - ▹ Behavior,
- And
    - Vice:
        - ▹ Versa.
- And
    - So
        - ▹ Let:

$$\mathscr{A}_{G_1}, \quad \mathscr{A}_{G_2} \quad \in \quad C_{G_1} \quad \cap \quad C_{G_2},$$

and

$$\mathscr{A}_{H_1}, \quad \mathscr{A}_{H_2} \quad \in \quad C_{H_1} \quad \cap \quad C_{H_2},$$

- Such that:

$$|\mathscr{A}_{G_1}| \quad = \quad |\mathscr{A}_{H_1}|, \qquad \textit{and} \qquad |\mathscr{A}_{G_2}| \quad = \quad |\mathscr{A}_{H_2}|.$$

- Then
    - Since:

$$C_{G_1} \quad \cap \quad C_{G_2} \quad = \quad C_{H_1} \quad \cap \quad C_{H_2},$$

we see that,

- There
    - Will:
        - ▹ Be

a

- Mapping

    - ○ Between:

$$\text{“\textit{the boundary-conditions}”}$$

- In:

$$C_{G_1} \quad \cap \quad C_{G_2} \qquad\qquad \textit{and} \qquad\qquad C_{H_1} \quad \cap \quad C_{H_2}.$$

- And
    - ○ So
        - ▷ If:

$$\mathscr{A}_{G_1} \quad \succ \quad \mathscr{A}_{G_2},$$

- Then:

$$\mathscr{A}_{H_1} \quad \succ \quad \mathscr{A}_{H_2}.$$

- And so
    - ○ From
        - ▷ This,
            - – We see that:

$$|\mathscr{A}_{G_1} \quad \oplus \quad \mathscr{A}_{G_2}| \qquad = \qquad |\mathscr{A}_{H_1} \quad \oplus \quad \mathscr{A}_{H_2}|.$$

- And
    - ○ Similarly,
        - ▷ For:

$$(\|, \quad \uplus), \qquad\qquad (\circlearrowright, \quad \otimes).$$

- And so
    - ○ From
        - ▷ This,
            - – We see that:

$$\Psi(C_{G_1} \quad \cup \quad C_{G_2}) \quad = \quad \Psi(C_{H_1} \quad \cup \quad C_{H_2}),$$

- And
  - So:

$$\Psi(C_G) \quad = \quad \Psi(C_H),$$

- Where:

$$\begin{aligned} C_G &= C_{G_1} \cup C_{G_2} \cup \ldots \cup C_{G_u}, \\ C_H &= C_{H_1} \cup C_{H_2} \cup \ldots \cup C_{H_u}. \end{aligned}$$

- Also
  - From
    - ▹ Lemma 9,
      - – We see that:

$$C_G \in G^\star \qquad \textit{and} \qquad C_H \in H^\star.$$

- And so
  - From
    - ▹ Theorem 2.1,
      - – We see that:

$$G \cong H.$$

Lemma 10. *If:*

$$M(D(G), \quad D(H))$$

- *Then:*

$$G \cong H.$$

- See
  - McKay [3],

        - For:
            - Proof,

when

- The
    - Number
        - Of:
            - Vertices

is

- Less
    - Than:

$$9.$$

Theorem 3.1. *Ulam's-Conjecture*

- *Is*
    - *True*
        - *For:*

"finite-connected-graphs"

with

- *No:*

self-loops and multi-edges.

Appendix

Definition 3. Let:

$$\text{`}\subset_b\text{'}$$

be

- A subset
    - Of:

$$S_G^\star \quad \times \quad S_G^\star.$$

- Then
    - We
        - ▹ Say:

$$C_a \quad \subset_b \quad C_b$$

if

- And
    - Only:
        - ▹ If

for

- All:

$$\mathscr{A}_{a_1}, \quad \mathscr{A}_{a_2} \quad \in \quad C_a,$$

$$I(\mathscr{A}_{a_1}) \quad \neq \quad I(\mathscr{A}_{a_2}),$$

$$T(\mathscr{A}_{a_1}) \quad = \quad T(\mathscr{A}_{a_2})$$

implies

- That
    - There

            - ▹ Exists:

$$\mathscr{A}_{b_1}, \quad \mathscr{A}_{b_2} \quad \in \quad C_b,$$

- Such that:

$$\begin{aligned} |\mathscr{A}_{a_1}| &= |\mathscr{A}_{b_1}|, \\ |\mathscr{A}_{a_2}| &= |\mathscr{A}_{b_2}|, \\ I(\mathscr{A}_{b_1}) &\neq I(\mathscr{A}_{b_2}), \\ T(\mathscr{A}_{b_1}) &= T(\mathscr{A}_{b_2}). \end{aligned}$$

Definition 4. Let:

$$`\subset_c'$$

be

- A subset
    - ○ Of:

$$S^\star_G \quad \times \quad S^\star_G.$$

- Then
    - ○ We
        - ▹ Say:

$$C_a \quad \subset_c \quad C_b$$

if

- And
    - ○ Only:
        - ▹ If

for

- All:

$$\begin{aligned}\mathscr{A}_{a_1},\quad \mathscr{A}_{a_2} \quad &\in \quad C_a,\\ I(\mathscr{A}_{a_1}) &= I(\mathscr{A}_{a_2}),\\ T(\mathscr{A}_{a_1}) &= T(\mathscr{A}_{a_2})\end{aligned}$$

implies

- That
    - There
        - ▹ Exists:

$$\mathscr{A}_{b_1},\quad \mathscr{A}_{b_2} \quad \in \quad C_b,$$

- Such that:

$$\begin{aligned}|\mathscr{A}_{a_1}| &= |\mathscr{A}_{b_1}|,\\ |\mathscr{A}_{a_2}| &= |\mathscr{A}_{b_2}|,\\ I(\mathscr{A}_{b_1}) &= I(\mathscr{A}_{b_2}),\\ T(\mathscr{A}_{b_1}) &= T(\mathscr{A}_{b_2}).\end{aligned}$$

Definition 5. Let:

$$\text{‘}\subset_d\text{’}$$

be

- A subset
    - Of:

$$S_G^\star \quad \times \quad S_G^\star.$$

- Then
    - We

        - ▹ Say:

$$C_a \quad \subset_d \quad C_b$$

if

- And
    - ○ Only:
        - ▹ If

for

- All:

$$\mathscr{A}_{a_1}, \quad \mathscr{A}_{a_2} \quad \in \quad C_a,$$

$$I(\mathscr{A}_{a_1}) \quad = \quad T(\mathscr{A}_{a_2}),$$

$$T(\mathscr{A}_{a_1}) \quad \neq \quad I(\mathscr{A}_{a_2})$$

implies

- That
    - ○ There
        - ▹ Exists:

$$\mathscr{A}_{b_1}, \quad \mathscr{A}_{b_2} \quad \in \quad C_b,$$

- Such that:

$$|\mathscr{A}_{a_1}| \quad = \quad |\mathscr{A}_{b_1}|,$$

$$|\mathscr{A}_{a_2}| \quad = \quad |\mathscr{A}_{b_2}|,$$

$$I(\mathscr{A}_{b_1}) \quad = \quad T(\mathscr{A}_{b_2}),$$

$$T(\mathscr{A}_{b_1}) \quad \neq \quad I(\mathscr{A}_{b_2}).$$

Definition 6. Let:

$$\text{`}\subset_e\text{'}$$

be

- A subset
    - ○ Of:

$$S_G^\star \quad \times \quad S_G^\star.$$

- Then
    - ○ We
        - ▷ Say:

$$C_a \quad \subset_e \quad C_b$$

if

- And
    - ○ Only:
        - ▷ If

for

- All:

$$\mathscr{A}_{a_1}, \quad \mathscr{A}_{a_2} \qquad \in \qquad C_a,$$

$$I(\mathscr{A}_{a_1}) \quad = \quad T(\mathscr{A}_{a_2})$$

$$T(\mathscr{A}_{a_1}) \quad = \quad I(\mathscr{A}_{a_2})$$

implies

- That
    - ○ There
        - ▷ Exists:

$$\mathscr{A}_{b_1}, \quad \mathscr{A}_{b_2} \qquad \in \qquad C_b,$$

- Such that:

$$\begin{aligned} |\mathscr{A}_{a_1}| &= |\mathscr{A}_{b_1}|, \\ |\mathscr{A}_{a_2}| &= |\mathscr{A}_{b_2}|, \\ I(\mathscr{A}_{b_1}) &= T(\mathscr{A}_{b_2}), \\ T(\mathscr{A}_{b_1}) &= I(\mathscr{A}_{b_2}). \end{aligned}$$

Definition 7. Let:

$$\text{`}\subset_f\text{'}$$

be

- A subset
  - ○ Of:

$$S_G^\star \quad \times \quad S_G^\star.$$

- Then
  - ○ We
    - ▷ Say:

$$C_a \quad \subset_f \quad C_b$$

if

- And
  - ○ Only:
    - ▷ If

for

- All:

$$\mathscr{A}_{a_1}, \quad \mathscr{A}_{a_2} \quad \in \quad C_a,$$

$$\begin{aligned} I(\mathscr{A}_{a_1}) &\neq I(\mathscr{A}_{a_2}), \\ T(\mathscr{A}_{a_1}) &\neq T(\mathscr{A}_{a_2}), \\ I(\mathscr{A}_{a_1}) &\neq T(\mathscr{A}_{a_2}), \\ T(\mathscr{A}_{a_1}) &\neq I(\mathscr{A}_{a_2}) \end{aligned}$$

implies

- That
    - ○ There
        - ▷ Exists:

$$\mathscr{A}_{b_1}, \quad \mathscr{A}_{b_2} \quad \in \quad C_b,$$

- Such that:

$$\begin{aligned} |\mathscr{A}_{a_1}| &= |\mathscr{A}_{b_1}|, \\ |\mathscr{A}_{a_2}| &= |\mathscr{A}_{b_2}|, \\ I(\mathscr{A}_{b_1}) &\neq I(\mathscr{A}_{b_2}), \\ T(\mathscr{A}_{b_1}) &\neq T(\mathscr{A}_{b_2}), \\ I(\mathscr{A}_{b_1}) &\neq T(\mathscr{A}_{b_2}), \\ T(\mathscr{A}_{b_1}) &\neq I(\mathscr{A}_{b_2}). \end{aligned}$$

References

[1] Hopcroft, John E.; Ullman, Jeffrey D., *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, 1979.

[2] Harary, F., *Graph Theory*, Reading, MA: Addison-Wesley, 1994.

[3] McKay, B. D., *Small graphs are reconstructible*, Australas. J. Combin., 15 (1997), 123 - 126.